

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



TRABAJO FIN DE GRADO

**INTEGRACIÓN DE REDES SOCIALES A UNA
PÁGINA WEB DE SERVICIOS**

Roberto Rozalén Calvo

Tutor: Javier Calleja Arregui

Ponente: Idoia Alarcón Rodríguez

MAYO 2014

RESUMEN

El objetivo de este proyecto, estimado en 300 horas de trabajo, es crear un módulo de carácter social para mejorar la interacción de los usuarios en un servicio web. Dicho módulo tiene la función de, por un lado mejorar la usabilidad de la aplicación frente al usuario; y por otro, permitir una interacción más sencilla y de mayor calidad entre los usuarios tanto dentro como fuera de la aplicación.

Para ello, se ha utilizado la API proporcionada por la red social Facebook para desarrollar un conjunto de funcionalidades que podemos agrupar en los siguientes objetivos:

1. Creación de una entidad para un tipo de usuario hasta ahora anónimo, permitiendo al mismo comunicarse con el resto de usuarios de una forma más efectiva y personal dentro de la aplicación.
2. Identificación automática utilizando la cuenta de Facebook del usuario, lo que permite reducir el proceso de identificación y ofrecerle nuevas funcionalidades relacionadas con dicha red social.
3. Permitir a los usuarios utilizar datos de Facebook en nuestra aplicación. Algunos ejemplos son invitar a amigos de Facebook, utilizar su imagen de perfil como avatar o compartir fotos de forma automática entre Facebook y la aplicación.
4. Dar la posibilidad a los usuarios de compartir la actividad que llevan a cabo dentro de la aplicación con sus amigos de Facebook, lo que además otorga publicidad a la misma.

Dichas funcionalidades han sido implementadas e integradas en el servicio web que proporciona la empresa Zankyou Ventures. Dicho servicio cuenta con 120.000 visitas diarias y 260.000 usuarios, por lo que se ha podido dar un uso real del trabajo y obtener resultados tangibles.

Tras la implementación de las mismas, se ha observado un aumento del 20% de la cantidad de usuarios que han ligado su cuenta de Facebook a la aplicación, por lo que se ha considerado que el proyecto ha tenido buenos resultados. Las funcionalidades que no han podido ser implementadas en el tiempo previsto se han establecido como trabajo futuro, al igual que otras muchas funcionalidades propuestas a partir del trabajo desarrollado.

The target of this project, estimated to be 300 working hours, is to create a social module to improve the interaction between users in a web service. The main function of this module is to improve the users usability in our application and also to enable a simpler and better interaction between users, both inside and outside our application.

To achieve this, we have used the API provided by the social network Facebook to develop a set of features, which can be arranged into the following:

1. Creation of an entity for a type of user who was anonymous until now, letting him to communicate with the other users in a more effective and personal way inside the application.
2. Automatic user identification using his Facebook account, which lets us to lighten the identification process and offer him new functionalities related to the given social network.
3. Let the users to use their Facebook data on our application. Some examples are: to invite Facebook friends, use their Facebook profile image as avatar or share photos automatically between Facebook and the application.
4. Provide the possibility to users of sharing the activity that they are carrying out inside our application with their Facebook friends, which also does some advertising of the web.

The features described above have been developed and integrated in the web service provided by the company Zankyou Ventures. This service has 120.000 daily visits and 260.000 users. Therefore, we have given a real use to this work and we have obtained tangible results of it.

After the development of the given features, we have perceived an increase of 20% of the amount of users who have bound their Facebook account to our application. Therefore, we consider that the project has been successful. The features that have not been implemented on schedule have been established as future work, as well as other features proposed from the developed work.

ÍNDICE DE CONTENIDO

1	INTRODUCCIÓN	1
1.1	Necesidades y relevancia	1
1.2	Objetivos	2
1.3	Análisis del estado del arte	3
1.3.1	Servicios existentes	3
1.3.2	Herramientas de desarrollo	7
1.4	Conclusiones.....	9
2	DEFINICIÓN DEL PROYECTO	10
2.1	Objetivos generales y específicos	10
2.2	Acotación.....	11
2.3	Hipótesis de partida	12
2.3.1	Estado inicial.....	12
2.3.2	Objetivo inicial.....	14
2.4	Técnicas y tecnologías utilizadas.....	16
2.4.1	Tecnologías utilizadas.....	16
2.4.2	Patrones de diseño.....	16
2.5	Metodología utilizada	17
3	ANÁLISIS DE REQUISITOS	18
3.1	Requisitos funcionales.....	18
3.2	Requisitos no funcionales	25
4	IMPLEMENTACIÓN	27
4.1	Diseño e implementación	27
4.1.1	Identificación de usuarios (RF1)	27
4.1.2	Identificación de invitados (RF2).....	30
4.1.3	Barra de invitado (RF3).....	33
4.1.4	Adición de invitados desde otros servicios (RF4)	35
4.1.5	Publicación en Facebook (RF5).....	37
4.1.6	Datos de usuarios (RF6).....	41
4.1.7	Galería de imágenes (RF7).....	44
4.1.8	Otras funcionalidades y problemas encontrados	45
4.2	Validación	48
4.3	Aplicación a un caso real	49
4.4	Evaluación de resultados.....	50

5	CONCLUSIONES Y LÍNEAS FUTURAS	53
6	BIBLIOGRAFÍA	55
7	BIBLIOGRAFÍA COMPLEMENTARIA.....	55

ÍNDICE DE FIGURAS

Figura 1: iCasei	3
Figura 2: Series.ly	4
Figura 3: Pinterest	5
Figura 4: Goodreads	6
Figura 5: API de Facebook	7
Figura 6: API de Twitter	7
Figura 7: Avatars.io.....	8
Figura 8: Enlazar una cuenta con Facebook.....	29
Figura 9: Cuenta enlazada con Facebook.....	29
Figura 10: Paneles de identificación de invitados	30
Figura 11: Panel de identificación de invitados (versión sin Javascript)	31
Figura 12: Javascript de comparación de usuarios	31
Figura 13: Evento con barra de invitado	33
Figura 14: Diseños de barra de invitado	33
Figura 15: Versión móvil de la barra	34
Figura 16: Comparativa de importación de contactos	35
Figura 17: Comparativa del bloque de importación de contactos.....	35
Figura 18: Formulario de contactos	36
Figura 19: Facebook <i>Share</i>	37
Figura 20: Facebook <i>Story</i>	37
Figura 21: Objetos personalizados	38
Figura 22: Acciones personalizadas.....	38
Figura 23: Historias personalizadas.....	39
Figura 24: Avatar de invitado	41
Figura 25: Sección de imagen.....	41
Figura 26: Avatar de usuario obtenido mediante Facebook.....	42
Figura 27: Sección de administración del invitado.....	43
Figura 28: Comentar una foto	44
Figura 29: Estadísticas generales de la aplicación.....	50
Figura 30: Número total de inicios de sesión.....	51
Figura 31: Llamadas diarias a la API de Facebook.....	51
Figura 32: Tasa de error de las llamadas a la API.....	52

ÍNDICE DE TABLAS

Tabla 1: Requisitos no funcionales.....	26
Tabla 2: Validación de apartados de requisitos funcionales.....	48

GLOSARIO

Evento: Sección asociada a una pareja y destinada a mostrar datos de la boda de la misma.

Usuario: Cualquier cliente de la aplicación, independientemente de su rol.

Pareja: Par de usuarios de la aplicación que tienen un evento asociado.

Invitado: Cualquier usuario que pueda acceder a un evento, a excepción de la pareja.

Spider: Software destinado a extraer información de una página mediante métodos de *crawling*.

API: *Application Programming Interface*. Métodos que ofrece cierta biblioteca para ser utilizado de forma externa.

Servicios: Referido exclusivamente a servicios ofertados mediante la web.

PALABRAS CLAVE

Identificación de usuarios	<i>Users identification</i>
Red social	<i>Social network</i>
Servicio web	<i>Web service</i>
PHP	<i>PHP</i>
Javascript	<i>Javascript</i>
Zankyou Weddings	<i>Zankyou Weddings</i>
API de Grafos de Facebook	<i>Facebook Graph API</i>
Facebook	<i>Facebook</i>

1 INTRODUCCIÓN

Este documento recoge el proyecto implementado por el estudiante para la empresa Zankyou Ventures en el plazo entre Enero y Junio del 2014. Dicho proyecto se ha llevado a cabo en un periodo de aproximadamente 300 horas y ha consistido en la integración de varias funcionalidades de carácter social y relativas a la API de Facebook en la página web de servicios www.zankyou.com. Tanto este documento como la implementación de las funcionalidades presentes en el mismo han sido realizadas de forma íntegra por el alumno.

El documento presenta la siguiente **estructura**:

En el resto de esta sección de **introducción** se introducirán los objetivos de este proyecto, las necesidades que los han impulsado y la relevancia del mismo. También se llevará a cabo un análisis del estado del arte en el que se mostrarán ejemplos de aplicaciones existentes para el trabajo que se llevará a cabo en el transcurso de este proyecto.

En la siguiente sección se **definirán los objetivos del proyecto**, y técnicas y tecnologías que se usarán en el mismo.

En la tercera sección se llevará a cabo un **análisis de requisitos**, tanto funcionales como no funcionales, de los que se compondrá este proyecto.

A continuación tiene lugar la sección de **implementación**, en que se documentará el trabajo realizado y se llevará a cabo la validación y evaluación de resultados del mismo.

Por último, se encuentran las **conclusiones y líneas futuras** de este proyecto, y la **bibliografía** utilizada en el transcurso del mismo.

1.1 Necesidades y relevancia

En la actualidad existe un gran número de servicios web cuyo objetivo es ofrecer una interacción de cualquier tipo entre los usuarios. Sin embargo, aún existe un gran número de webs que no hacen uso de todo el potencial con el que cuentan las herramientas de desarrollo actuales.

Esto trae consigo que los usuarios tengan que atravesar webs con una nula usabilidad para comunicarse de una forma poco eficiente, marcada más por la arquitectura de dichas webs que por la necesidad de los mismos.

Por ello, en este proyecto se pretende hacer uso de algunas de las herramientas más actuales para mejorar la parte social de una web de servicios y establecer un ejemplo del correcto uso de dichas herramientas.

Consideraremos que el trabajo ha tenido éxito si se aprecian cambios positivos relacionados con la interacción de los usuarios con la web, o si aumenta el número de usuarios atraídos por la implementación de estos nuevos cambios.

1.2 Objetivos

El propósito general de este proyecto es mejorar la interacción entre los usuarios que participan en un servicio web, mejorando así la calidad del servicio ofertado. Junto a este objetivo también se espera que aumente la publicidad de nuestro servicio, y por lo tanto el número de usuario.

Para ello se han establecido una serie de objetivos destinados a mejorar cada uno de los aspectos relacionados con dicha interacción. Son los siguientes:

- Facilitar la identificación de los usuarios ofreciendo un sistema de *login* automático mediante una red social como Facebook.
- Ofrecer una entidad a un tipo de usuario que era hasta ahora anónimo, permitiéndole participar de forma activa en la web.
- Permitir a los usuarios añadir a sus amigos de Facebook como invitados, y extraer información de ellos, como sus avatares.
- Facilitar a los usuarios la forma de compartir lo que ocurre en nuestra web de una forma automática y personalizada utilizando la API de Facebook.

El resultado ha sido muy satisfactorio, ya que ha aumentado un 20% la cantidad de usuarios que se conectan mediante Facebook a nuestra aplicación, lo que nos permite presentar a los mismos nuevas funcionalidades destinadas a mejorar la interacción entre ellos tanto dentro de la aplicación como en el exterior de la misma.

1.3 Análisis del estado del arte

Se ha llevado a cabo un análisis para determinar el estado de un conjunto de servicios web cuyo objetivo principal es el de proporcionar un servicio de carácter social para el usuario.

Han quedado excluidas de dicho análisis las redes sociales propiamente dichas, ya que no se pretende aumentar la funcionalidad ni valorar la utilidad de las mismas, sino añadir un módulo social a una web de carácter diferente.

Durante el análisis se han tenido en cuenta tanto las funcionalidades que ofrece cada servicio como criterios de usabilidad y accesibilidad de cara al usuario.

1.3.1 Servicios existentes

Se han analizado las siguientes páginas de servicios web:

Site de Casamento iCasei

Elegida mejor web de boda de Brasil en 2007. La web permite crear una lista de boda y almacenar fotos y eventos de la boda en la web. [1]



Figura 1: iCasei

Elegida por similitud con nuestro servicio y por su uso de las herramientas de Facebook.

Ofrece un servicio de lista de boda, que permite a la pareja elaborar dicha lista para permitir a los invitados hacer los regalos transfiriendo a la pareja una cantidad establecida relativa a cada regalo. También permite a la pareja llevar a cabo una página a modo de blog (dentro del dominio de dicha web), que puede utilizar para mostrar a los invitados novedades relativas a la boda, subir fotografías o establecer la agenda de la misma.

Dicha web contiene también un módulo de funcionalidades ofrecidas por Facebook que permite generar una página de Facebook automáticamente mostrando un resumen, los últimos cambios realizados (comentarios, fotos y nuevas entradas) y un enlace a la página del evento.

Sin embargo, no ofrece más funcionalidades de carácter social, ni hace otro uso de las herramientas que ofrece Facebook. Además se podría ampliar información mostrada por la página de Facebook para no mostrar únicamente un resumen, sino proporcionar un clon del evento dentro de la propia red social. Dicha página se actualizaría automáticamente cuando se actualizase el evento e incluso se podría permitir que los cambios de dicha página se reflejasen en el propio evento de la página web.

Series.ly, Tu televisión social

Ofrece una red social basada en el interés de los usuarios por series y películas. Permite compartir enlaces de contenido audiovisual y críticas y comentarios sobre el mismo. [2]



Figura 2: Series.ly

Elegida por su interés social y de cooperación entre usuarios en un campo diferente como el del contenido multimedia.

Dicha web utiliza un sistema comunitario tanto para ofrecer contenido como para moderar el mismo. Los usuarios son los que suben contenido, lo moderan y comentan, permitiendo así un crecimiento mucho mayor y más rápido de la cantidad de contenido que ofrece la página. Para favorecer dicho sistema, la web utiliza un sistema de karma, asignando puntos a los usuarios al realizar tareas que favorezcan esta compartición de contenido, que podrán utilizar para visualizar el propio contenido o tener ventajas como eliminar tiempos de espera o publicidad.

Permite a su vez que los usuarios interaccionen entre ellos, comentando el contenido o recomendándolo a otros usuarios. También les permite mantener conversaciones mediante un sistema de mensajes privados.

Por otro lado, no integra los servicios de ninguna red social y no facilita la publicidad de la misma, ya que la página es privada y el registro en la misma requiere de una invitación de un usuario actualmente registrado. Gracias a ello, incorpora un sistema de identificación propio, de modo que no permite realizar dicha identificación de forma automática con el uso de alguna otra cuenta de correo o red social.

Pinterest: *The visual discovery tool*

Permite coleccionar ideas sobre un tema o proyecto en forma de imágenes. Su objetivo es ofrecer ideas y servir de inspiración a la hora de embarcarse en un proyecto o interés. [3]



Figura 3: Pinterest

Elegida también por su objetivo estrictamente social, esta página de uso gratuito permite visualizar contenido multimedia (en su mayor parte imágenes) subido por los usuarios.

Permite ver contenido subido por otros usuarios y establecer categorías para la visualización de dicho contenido. Cuenta con una gran integración con Facebook, pero no cumple una gran cantidad de requisitos de usabilidad básicos ni muchas de las recomendaciones que la propia red social aporta a la hora de utilizar su API. Entre otras, las siguientes:

- El pop-up para conectarse con Facebook muestra dos pantallas además del formulario de identificación de usuario. Una de ellas pide al usuario permisos de lectura de su Facebook y la otra, permisos de escritura. Ambas cosas dificultan la tarea de introducir al usuario a la aplicación mediante Facebook, ya que es muy probable que el usuario cancele la acción al ver todos los requisitos que pide la aplicación.
- La aplicación comparte automáticamente en la cuenta de Facebook del usuario cierto contenido, sin preguntar por el mismo. Esto genera una desconfianza por parte del usuario hacia la aplicación que puede dar lugar al cese del uso de la misma.

Goodreads

Fusiona las funcionalidades de una red social con un **potente un sistema de recomendación** para ofrecer a los usuarios la posibilidad de encontrar libros acordes con sus gustos, o ver los libros leídos por sus amigos. [4]

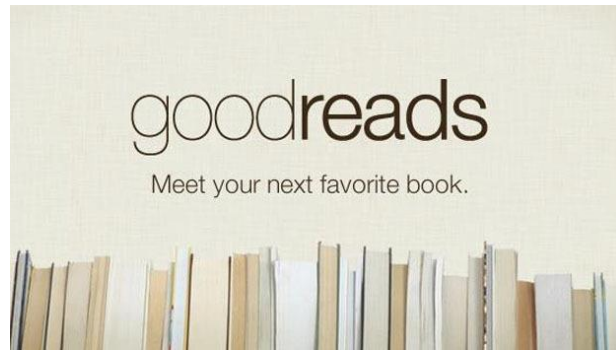


Figura 4: Goodreads

Dicha web es una elección interesante, ya que utiliza una herramienta de la *Graph Api* de Facebook muy potente y poco utilizada: *Facebook custom stories*.

Esta herramienta permite compartir de forma automática logros o acciones que lleva a cabo el usuario con sus amigos de Facebook publicando una nota en su muro con un título y una imagen personalizados (por ejemplo "*Roberto ha leído el libro La Sombra del Viento en Goodreads*") y una pequeña nota de contenido en forma de texto. Se utilizará esta herramienta para llevar a cabo el **RF5** del apartado **3.1** y se hablará de ella detalladamente en la sección de implementación de dicho requisito.

También, además de facilitar la posibilidad de identificación con Facebook, permite identificar de forma automática al usuario en el momento en el que entra en la página si se ha identificado anteriormente con dicha red social y tiene una sesión abierta con la misma.

Por otro lado, este dominio web también incumple varias de los requisitos no funcionales que Facebook propone para el uso correcto de su API:

- Al identificar a un nuevo usuario, la página pide todos los permisos que necesita, tanto de lectura como de escritura, abriendo dos ventanas de diálogo adicionales. Esto puede desencadenar en un considerable número de usuarios que no acepten los permisos, no se identifiquen, o dejen de navegar por la página.
- Los botones que disparan las acciones propias de Facebook siguen el diseño de la web en vez de el diseño recomendado por Facebook y es difícil para el usuario determinar si dichas acciones tienen relación con Facebook. Se ganaría mucho en usabilidad de la web si se utilizara el diseño predeterminado de Facebook con todo lo relacionado con sus acciones o al menos se dejara evidencia de esta relación.

1.3.2 Herramientas de desarrollo

A su vez, se ha valorado el uso de las siguientes herramientas de desarrollo para ser utilizadas en la implementación de este proyecto:

Facebook API

Su uso permite a cualquier servicio comunicarse con Facebook para obtener información o modificar cualquier dato perteneciente a cualquier elemento de Facebook. Ya sea un usuario, página, grupo, etc. [5]



Figura 5: API de Facebook

Permite llevar a cabo tareas como la identificación del usuario y la elección de privacidad y visibilidad de una forma sencilla utilizando la cuenta del usuario de dicha web social.

Es la herramienta más potente de la que dispone es la Graph API, que facilita cualquier dato en un formato Json como respuesta a una simple llamada GET. Utiliza un esquema basado en grafos que permite acceder a la información de una forma cómoda sin necesidad de hacer búsquedas o peticiones complejas a una base de datos.

Consta de varios SDK, entre ellos el SDK de Javascript que utilizaremos durante el desarrollo.

Twitter API

Facilita la obtención de datos de Twitter y permite a su vez postear mensajes con la identidad de un usuario a dicha web. [6]



Figura 6: API de Twitter

Se podría dar uso a dicha API para obtener avatares de usuarios y para permitir que los mismos comenten detalles o cambios acerca del evento en Twitter. También permite llevar a cabo la identificación del usuario utilizando la cuenta de dicha red social, por lo que se estimará si merece la pena integrar una funcionalidad que permita al usuario llevar a cabo su identificación utilizando diferentes redes sociales.

Avatars.io

Encapsula las API de las principales redes sociales para proporcionar un avatar de usuario de una forma cómoda y sencilla. Para ello es necesario facilitar a dicho servicio el nombre de usuario de la red social en cuestión o su identificador de usuario. [7]



Figura 7: Avatars.io

También tiene integración con Gravatar, que es un sistema presente actualmente en el servicio web que queremos ampliar y utilizado para mostrar avatares de usuario en caso de que el usuario no haya especificado un avatar y quiera utilizar su identidad de Gravatar de cara a los demás usuarios.

En un principio, se había planteado su integración en este proyecto por las razones anteriores. Sin embargo, hemos decidido no utilizarlo, ya que para ello es necesario pedir a los usuarios de nuestro servicio sus identificadores o nombres de usuario de las distintas redes sociales y podría conllevar a una respuesta o un efecto negativo por parte de los usuarios. Por otro lado, podemos llevar a cabo dicha funcionalidad directamente utilizando la API proporcionada por Facebook.

1.4 Conclusiones del análisis del estado del arte

Queremos añadir funcionalidad de carácter social web de servicios (Zankyou.com) [8]. Dicha web cuenta con dos tipos de usuarios, pero únicamente tiene un sistema de identificación con uno de ellos. La interacción entre dichos usuarios funciona de un modo prácticamente unilateral, a modo de blog. Por último, el servicio cuenta con una página de Facebook para darse a conocer que está separada del propio servicio, sin embargo los usuarios no pueden subir a Facebook ni a ninguna otra red social las actividades y aportaciones que llevan a cabo dentro del evento de Zankyou.

A partir del estudio detallado de distintos servicios y herramientas existentes, como se ha analizado a lo largo de este capítulo, se pueden extraer una serie de conclusiones útiles para el desarrollo propuesto de este trabajo:

- i. La funcionalidad potencial que se ha podido determinar gracias al análisis del estado del arte es la posibilidad de identificar a ambos tipos de usuarios, y además facilitar a los mismos dicha identificación utilizando sus cuentas de Facebook. Cada usuario tendrá una entidad que se remarcará en el servicio, con funcionalidades tales como mostrar el avatar de usuario en sus actividades y mensajes, o la posibilidad de etiquetarse a sí mismos o a otros en fotos que hayan subido a nuestro servicio.
- ii. Por otro lado, gracias al análisis de los servicios iCasei y Goodreads hemos podido comprobar que utilizando la API de Facebook podemos publicar tanto un resumen de la actividad del usuario como las actualizaciones que sufre un determinado evento y precisar el nivel de privacidad que quiere el usuario en todo momento. Esto permitirá que los usuarios de este servicio hagan referencias al mismo por Facebook, incrementando así la publicidad del servicio.
- iii. Finalmente, el análisis de Pinterest y otras webs ha dejado clara la importancia de los aspectos no funcionales dedicados a mantener la usabilidad del producto y la confianza del usuario en nuestra aplicación a la hora de utilizar cualquier API social. En posteriores apartados se determinarán los requisitos no funcionales a tener en cuenta durante el desarrollo de este módulo.

2 DEFINICIÓN DEL PROYECTO

2.1 Objetivos generales y específicos

Objetivos generales:

El principal objetivo es ampliar las funcionalidades ofertadas por el dominio web de servicios de Zanky you [8] para ofrecer un mayor nivel de funcionalidad social a los clientes. Para ello se deberá facilitar una entidad a los usuarios, (que hasta ahora constituían un elemento prácticamente anónimo), y ampliar las acciones que se pueden realizar para mejorar la interacción entre los mismo utilizando las herramientas de las que dispone la empresa Zanky you Ventures. Dichas nuevas funcionalidades, que serán implementadas en este proyecto, pueden agruparse dentro de los objetivos expuestos a continuación.

Objetivos específicos:

- i. Utilizar Facebook para identificar al tipo de usuario "pareja", que hasta este momento utilizaba un sistema de identificación propio de la web.
- ii. Identificar al tipo de usuario "Invitado", que no cuenta con identificación, e introducir sus datos en el sistema existente.
- iii. Mostrar datos de los invitados, como su nombre y avatar, en las acciones de las que forme parte.
- iv. Utilizar amigos de Facebook para rellenar perfiles de invitado del evento, pudiendo incluir varios invitados al mismo tiempo.
- v. Crear un "feed" automático en Facebook acerca del evento de la pareja.
- vi. Compartir fotos entre Facebook y Zanky you, ofreciendo la posibilidad de subir dichas imágenes en la máxima calidad posible de una web a otra.
- vii. Ofrecer al usuario la posibilidad de habilitar, deshabilitar o establecer el nivel de privacidad de las acciones anteriores.

Dichos objetivos se utilizarán posteriormente para definir y ampliar los requisitos funcionales de este proyecto en el punto **3.1**.

2.2 Acotación

La implementación del módulo propuesto en este proyecto no incluirá los siguientes aspectos:

- **Traducción a idiomas:** Todos los aspectos funcionales implementados deben ser traducidos a los 6 idiomas presentes en el servicio web de Zankyou. Sin embargo, será el equipo de traducción el que se encargue de esta tarea, por lo que no será desarrollada en este módulo.
- **Diseño y maquetación:** El diseño de las vistas y elementos a implementar será proporcionado por Zankyou Ventures. Sí se llevará a cabo la integración de dichos elementos junto con las nuevas funcionalidades dentro de este proyecto.

Las siguientes funcionalidades no entrarán en el desarrollo de este proyecto, pero se plantearán para **futuras mejoras e implementaciones**:

- Crear una página de Facebook generada automáticamente por un evento que sea una muestra o resumen de dicho evento y permita acceder al mismo desde Facebook por medio de un enlace.
- Incluir, dentro cualquier foto perteneciente al álbum fotográfico de un evento, la capacidad de identificar “tag” a los invitados del mismo que aparezcan en la fotografía. Se definirán los tipos de usuario que pueden realizar esta acción y se importarán automáticamente los “tags” de las fotografías importadas desde Facebook a la aplicación.
- Incluir la posibilidad de llevar a cabo la identificación de los usuarios mediante otras redes sociales, como Google+ o Twitter.

Se hablará detalladamente de dichas funcionalidades en el apartado **5** del documento.

2.3 Hipótesis de partida

2.3.1 Estado inicial

Contamos con un negocio web dedicado a ofrecer una plataforma que permite facilitar a parejas que vayan a casarse la administración de la boda, ofreciendo facilidades para administrar la lista de invitados, la lista de regalos de boda y ofreciendo un espacio para que la pareja pueda informar a los invitados de cualquier evento relacionado con la boda o mostrar fotos a los mismos. Dicha plataforma web consta de las siguientes secciones:

Evento: Página asociada directamente con una boda, creada por la pareja y en la que puede subir fotos, escribir entradas o notas, y determinar el momento y lugar de todos los eventos (celebración, banquete, etc.) relacionados con la boda. En dicha página la pareja puede informar a los invitados de todo lo que va ocurriendo en el proceso de la boda y mostrar las fotos tomadas. A su vez, los invitados pueden escribir comentarios en cualquiera de las secciones del evento. Por último, también puede determinar la lista de bodas y permitir que los invitados elijan regalos de la misma y abonen una transferencia a los novios por el valor de los mismos. A continuación se detallan las secciones de las que se compone el evento:

- i. Lista de regalos: En ella la pareja elige plantea los regalos que quiere para la boda. Los invitados pueden elegir uno o varios regalos y efectuar una transferencia a la cuenta de la pareja por el valor de los mismos. La pareja también puede ofrecer a los invitados el realizar una contribución libre para la boda, o hacer una contribución destinada a una ONG. Por último, los novios pueden establecer una contribución para un viaje, marcar las diferentes etapas del mismo y el dinero que estiman que costará cada etapa para que los invitados decidan hacer una contribución destinada al viaje de novios. En dicho caso se muestra un mapa indicando las diferentes etapas del viaje.
- ii. Álbum de fotos: En ella la pareja puede subir álbumes de fotos y fotografías desde su equipo. Los invitados pueden ver y comentar las fotos, pero no pueden subir o modificar ninguna fotografía o álbum. La pareja puede descargar también dichas fotos en calidad normal o en la máxima calidad disponible (esta última opción únicamente se encuentra disponible para las parejas que han adquirido dicha funcionalidad de pago).

- iii. Agenda: En ella la pareja puede determinar el momento y lugar de cualquier evento relacionado (o no) con su boda. Cada evento puede contar con un mapa que indique también la localización del mismo.
- iv. Hoteles: Destinada a que la pareja pueda facilitar la dirección de hoteles próximos a cada evento de la boda. Esta sección cuenta con un único mapa que mostrará cada uno de los mapas seleccionados por la pareja.
- v. Blogs y páginas: El blog constituye una sección y en él la pareja puede escribir *posts* con comentarios y contenido multimedia como imágenes, vídeo, música o widgets. Los invitados pueden comentar cada *post* de forma individual. Por otro lado, las páginas constituyen *posts* que no se encuentran dentro del blog, sino en una sección propia. A pesar de ello, su contenido y funcionamiento es el mismo descrito anteriormente.
- vi. Libro de visitas: Esta sección permite a los invitados escribir cualquier comentario, canción o vídeo. Dicho contenido se mostrará por orden de llegada, uno detrás de otro, en una lista de comentarios.
- vii. Confirmar asistencia: En esta sección cualquier invitado puede confirmar su asistencia a la boda (en caso de ya haber sido identificado por la pareja), o pedir su participación a la boda en caso contrario. En todo caso, la pareja es la encargada de ver los invitados que han enviado una confirmación y admitirles o no en la boda. En el primer caso se sucederán otro tipo de acciones, como asignar una mesa al invitado, o mandarle un email de cualquier tipo.

Administración personal: En dicha página la pareja puede administrar las transferencias de la lista de bodas y la configuración personal del servicio web.

Administración de invitados: La página de administración de invitados permite añadir invitados al evento, administrar la asistencia de los mismos, mandar correos y peticiones de confirmación a los mismos y asignarles mesas.

A continuación se detallan las **principales usuarios** presentes en el portal web para una mejor comprensión de la funcionalidad a añadir:

- i. Pareja: Usuarios que contratan el servicio gratuito. Tienen asignado un evento y pueden tanto administrar sus preferencias de usuario como dar forma y administrar su evento. También pueden comprar diferentes funcionalidades que no están disponibles en la versión gratuita.
- ii. Invitado: Usuarios invitados por la pareja. Ellos pueden acceder de forma anónima al evento creado por los anteriores, en el que pueden escribir comentarios en los diferentes apartados del evento y elegir regalos para los novios, además de informarse de cualquier cosa relacionada con la boda que haya escrito la pareja.

Los invitados pueden escribir comentarios relacionados con el contenido en todas las secciones. Para escribir un comentario tienen que especificar un título del comentario, descripción, nombre y email del invitado, además de rellenar un *captcha* destinado a bloquear posibles *spiders* o *bots* automatizados.

2.3.2 Objetivo inicial

El principal objetivo es ofrecer a los dos tipos de usuarios (pareja e invitado) una mejor experiencia a la hora de comunicarse entre ellos. Para ello se llevarán a cabo mejoras basadas en los objetivos específicos mencionados en el punto 2.1. El cumplimiento de dicho objetivo llevará también consigo la consecución de los siguientes:

- **Aumento de participación** de ambos tipos de usuarios en el evento.
- **Aumento de la calidad del evento**, que desemboca en la futura compra o mantenimiento del mismo por parte de la pareja.
- **Aumento del porcentaje de usuarios** que ligan su cuenta **de Facebook** con nuestra aplicación con respecto al número total de usuarios la aplicación.
- **Publicidad gratuita en Facebook** gracias a los comentarios de los usuarios sobre la aplicación.

Para ello, se llevarán a cabo los siguientes cambios en la aplicación:

Creación del rol "invitado":

La figura de invitado que se mantiene en la aplicación es un mero indicador de las personas que acudirán a la boda, utilizado para facilitar la administración de la misma a la pareja. Se creará un rol completo para el invitado, **otorgando una entidad real a este tipo de usuarios y permitiendo que se identifiquen a la hora de entrar en la aplicación**. Dicho tipo de usuario tendrá también una sección de administración, que le permitirá modificar sus datos y establecer una imagen de avatar. Dicha imagen aparecerá en las acciones que lleve a cabo dicho usuario.

Aumento del valor de las acciones de los usuarios en el evento:

Se otorgará un mayor valor a dichas acciones, **mostrando a la pareja la actividad y participación** que han tomado lugar en su evento, e identificando a las personas que han llevado a cabo dicha actividad. A su vez se **mostrará la identidad de los usuarios** que realicen acciones en el evento, como compartir un comentario o hacer un regalo.

Conexión con Facebook:

Se ofertarán nuevas funcionalidades para aumentar el uso conjunto de dicha red social con nuestra aplicación. Entre ellas se encuentran el ofrecer un **sistema de identificación** sólido y automático, **extraer imágenes de perfil** de los usuarios para mostrarlas en el evento, **importar la información de amigos** de Facebook como participantes de la boda, o **compartir imágenes** entre Facebook y la página de evento de la pareja.

Publicación en Facebook:

Los usuarios podrán **generar comentarios automáticos en su tablón de Facebook** a medida que participan en eventos o acciones de la página de los novios. Dichos comentarios presentarán información del evento en el que participan y de las acciones realizadas. También ofrecerán un enlace a la página del evento y mostrarán una imagen relacionada con el mismo. Dicha tarea aumentará también el valor de las acciones realizadas en el evento.

En el **apartado 3** de este documento se especifica el desglose de las tareas necesarias para llevar a cabo los objetivos mencionados.

2.4 Técnicas y tecnologías utilizadas

2.4.1 Tecnologías utilizadas

Para el desarrollo de esta aplicación se han utilizado las mismas tecnologías adoptadas por la empresa, además de las herramientas de desarrollo y **APIs** necesarias para este módulo en concreto. Dichas tecnologías son:

- **Javascript y JQuery:** Utilizado para cualquier prestación necesaria en el cliente.
- **Ajax:** Llamado desde Javascript, para ofrecer una respuesta asíncrona al cliente.
- **PHP 5.3 y HTML 5:** Componen la base de la página web
- **CSS3:** Responsable del diseño de la página.
- **MySQL:** Utilizado en el Framework para acceder a la base de datos.
- **Zend Framework:** Framework de PHP que facilita tanto la estructura interna de la página como las llamadas al servidor y acceso a la base de datos.

También se han utilizado herramientas de desarrollo proporcionadas por la red social Facebook para su integración en páginas web, en particular el SDK para Javascript y llamadas desde el servidor a su API, que será atacada mediante métodos POST.

Facebook ofrece también un SDK para PHP, pero se decidió que no era necesario utilizarlo, ya que supliría las llamadas a la API y la única ventaja que ofrece frente a las anteriores es la capacidad de mantener el estado de la sesión llevando a cabo una identificación mediante Facebook.

2.4.2 Patrones de diseño

A su vez se ha hecho uso de los siguientes patrones de diseño:

Model View Controller (MVC): Utilizado para estructurar todo el servicio web, este patrón de diseño permite separar el almacenamiento de datos, la lógica de negocio y el código dedicado a mostrar dichos datos. Como consecuencia, cada acción creada en el controlador tiene asociada una vista a la misma.

Template: Dicho patrón de diseño es útil para definir el esqueleto de un algoritmo permitiendo crear subclases que puedan redefinir ciertos pasos de dicho algoritmo sin alterar la estructura del mismo. Se ha utilizado para empaquetar toda la funcionalidad encargada de mostrar datos a partir de la API de Facebook, ya que dichos métodos comparten la misma estructura y serán más fáciles de mantener si unificamos dicha estructura.

Singleton: Utilizado a la hora de llamar a un módulo que únicamente necesite inicializarse una sola vez. Dichas llamadas inicia el módulo o usa uno ya existente. Utilizado también para llevar a asegurar la identificación en Facebook en cada método que lo haya requerido.

Por último, se ha aplicado conocimiento de sistemas UNIX, Windows, acceso remoto a servidores y bases de datos para llevar a cabo las tareas mencionadas, ya dicho conocimiento es absolutamente necesario para operar en el entorno de trabajo.

2.5 Metodología utilizada

Debido a la naturaleza del proyecto, se ha utilizado **una metodología ágil con iteraciones semanales** para llevar a cabo el desarrollo e implementación del mismo. La finalidad de esta metodología ha sido añadir un requisito funcional completo, o varias tareas pertenecientes al mismo, en cada iteración.

Cada iteración se ha compuesto de las siguientes **fases**:

- i. **Análisis de requisitos:** A partir de las ideas iniciales, el equipo de desarrollo ha desglosado los requisitos necesarios de cada funcionalidad.
- ii. **Diseño:** La empresa ha proporcionado el diseño de cada solución propuesta, así como la maquetación de la misma en caso de haber sido necesaria.
- iii. **Codificación:** En esta fase se ha efectuado la integración de la maqueta y la codificación de cada requisito funcional. Esta tarea se ha llevado a cabo de forma íntegra por el alumno.
- iv. **Revisión:** Se ha llevado a cabo una fase de revisión al final de la codificación de cada tarea antes de unir el código al de desarrollo. También se han llevado a cabo dos fases de revisión y corrección de errores a la hora de subir el código a la web de test y a la web pública para minimizar los posibles errores presentes en la tarea. Dichas revisiones se han llevado a cabo por el equipo de desarrollo, incluido el propio alumno.
- v. **Documentación:** En esta última fase se ha documentado cada tarea en este documento y en la propia documentación de la empresa en el caso de haber sido necesario.

3 ANÁLISIS DE REQUISITOS

3.1 Requisitos funcionales

Se definirán en este apartado tanto los requisitos funcionales propiamente como aquellos relacionados con la interfaz de usuario debido al papel principal que juega ésta en la aplicación, que hace que la mayoría de estos requisitos se puedan considerar parte de la funcionalidad del sistema.

RF1 - Identificación de pareja mediante Facebook

(RF1.I) Identificación del rol de usuario "pareja" mediante Facebook.

Se deberá añadir un botón de identificación con Facebook en la barra superior de opciones de identificación. Dicho botón pedirá el permiso de "email de usuario", necesario para realizar nuestra identificación del usuario. Se guardará en la base de datos el identificador de Facebook del usuario para futuro acceso de datos de Facebook y no se permitirá que una cuenta de Facebook se ligue con varias cuentas de Zankyou al mismo tiempo.

(RF1.II) Comprobación dinámica de la cuenta de usuario.

El sistema comprobará si dicha cuenta de Facebook existe en el sistema de Zankyou. De ser así se conectará automáticamente.

De otro modo, comprobará si se encuentra en el sistema el correo adquirido mediante Facebook. En ese caso ligará el identificador de Facebook a dicho usuario y realizará una conexión automática como en el caso anterior.

Si no se diese ninguna de las anteriores, el sistema rellenará el formulario de inicio con los datos del usuario obtenidos mediante Facebook y será este último el que deba comprobarlo y enviarlo para evitar crear cuentas no deseadas de forma automática.

(RF1.III) Tratamiento de permisos de Facebook.

Se creará un método que permita encapsular la lógica de identificación y permisos de Facebook para que se pueda utilizar de forma sencilla y genérica en toda la aplicación.

(RF1.IV) Conexión con Facebook.

El sistema permitirá enlazar una cuenta de Zankyou con una cuenta de Facebook desde su sección de administración. También permitirá en el caso de haber un enlace con Facebook, eliminar dicho enlace existente.

RF2 - Identificación de invitados

(RF2.I) Pantalla de identificación para el rol de "invitado".

Se debe añadir un formulario de identificación para el rol de "invitado". Dicho formulario se mostrará en los eventos que requieran una identificación de invitados cuando se intente acceder a un evento por primera vez en un cliente web. En los eventos que requieran también contraseña, se mostrará primero el formulario encargado de la contraseña y a continuación se mostrará este nuevo formulario. Pedirá email y nombre del usuario para llevar a cabo un registro del usuario. La vista de RSVP o petición de identificación del invitado será pública y no necesitará de dicha autenticación.

(RF2.II) Chequeo dinámico de usuarios.

El anterior formulario permitirá únicamente en un principio introducir el email y comprobará mediante AJAX si coincide con el de un invitado o miembro de la pareja. En el primer caso incluirá automáticamente el nombre del invitado y permitirá tanto elegir otro de los invitados con dicho correo (ya que el sistema permite que varios invitados se identifiquen con la misma cuenta de correo, pero distintas identidades), como crear uno nuevo. En otro caso, indicará con un mensaje de error al usuario que dicho correo corresponde al de un miembro de la pareja, por lo que deberá identificarse en la sección de identificación de parejas o utilizar otro correo.

(RF2.III) Pantalla de acceso a evento privado.

Los eventos cuentan con un formulario de acceso en el caso de que el evento sea privado. Debe modificarse dicho formulario para seguir el diseño del evento y obtener también la identificación del invitado.

(RF2.IV) Captcha para evitar bots.

Con motivo de evitar el acceso o registro indiscriminado de *spiders* se debe añadir un *captcha* a la pantalla de identificación de invitado. Dicho campo se mostrará únicamente en las sesiones que no utilicen Javascript, y deberá ocultarse mediante Javascript en el resto.

(RF2.V) Almacenar datos de visitas de los invitados.

Se utilizará la herramienta **Google Analytics** (actualmente usada en la web para datos de la pareja) para obtener datos de invitados, como la frecuencia con la que visitan la página. Dicha información utilizará para proporcionar a los usuarios datos del acceso al evento. (ver **RF6.VI**)

(RF2.VI) Mostrar la URL del evento como URL canónica.

Debe cambiarse el atributo “URL canónica” dentro de los atributos de cabecera utilizados para identificar la página de cara a buscadores externos. En lugar de la URL designada a las acciones de identificación de invitado y evento con código de acceso, se mostrará en ambos casos la URL de la página principal del evento. Pueden darse dos posibles casos:

Si la URL pertenece al dominio de Zankyou, dicha URL seguirá la estructura: `http://` seguido del código del evento, seguido del dominio. Ej: <http://anaydavid.zankyou.com>

Si la URL es externa, dicho atributo estará coincidir con la URL de la página principal de dicho evento. Ej: <http://www.anaydavid.com>

RF3 - Barra de invitado

Se debe incorporar una barra en la parte inferior del evento que muestre el nombre del invitado para remarcar que está identificado en la aplicación. En caso de no estarlo, mostrará la posibilidad de hacerlo. Dicha barra se adaptará al tamaño del cliente en el que la página sea visualizada. Se crearán iconos de las secciones para la versión de móvil de la barra. La maquetación de la misma será proporcionada por la empresa.

(RF3.I) Incorporación de la barra de invitado al evento.

(RF3.II) Preferencias

Incorporación de una sección a la página de preferencias que permita habilitar o deshabilitar la barra de invitado. Dicha barra estará habilitada por defecto.

(RF3.III) Chat de ayuda.

Se incorporará el chat de ayuda, presente en el resto de la aplicación, en la barra a modo de botón. Dicho botón abrirá la ventana emergente utilizada por dicha aplicación de chat.

(RF3.IV) Control de sonido.

Se eliminará el reproductor de música de los eventos y se colocará en su lugar un botón en la barra para activar o desactivar el sonido del evento.

(RF3.V) Secciones.

La barra contendrá un enlace a las secciones destinadas a escribir un mensaje, subir una foto y hacer un regalo. Dichos enlaces aparecerán únicamente en el caso en el que las secciones mencionadas estén habilitadas y su privacidad sea pública. En caso de no estar habilitada ninguna de las secciones se ocultará la barra de invitado.

(RF3.V) Identidad del usuario.

Se mostrará el nombre del usuario para hacerle saber el nombre con el que se ha identificado en el evento. En caso de no haberse identificado, se brindará la posibilidad de hacerlo mostrando en la barra un enlace a la página de identificación de invitados.

(RF3.V) Botón de "Me gusta".

Se incorporará un botón de "Like" que apuntará a la URL canónica del evento.

RF4 - Permitir a la pareja invitar amigos de Facebook a la aplicación

En la sección "Añade un invitado", en la que ya existe funcionalidad que permite descargar varios contactos al mismo tiempo desde Hotmail y Excel, se añadirá la posibilidad de descargar contactos desde Facebook, Google+ y el correo de Gmail. En todas ellas se intentará recopilar todos los datos posibles sobre los invitados, como email, nombre, identificador de Facebook en caso de utilizar dicha red social, etc. También se añadirá información de qué servicio provienen dichos usuarios.

(RF4.I) Descargar los datos de invitados de Facebook.

(RF4.II) Descargar los datos de invitados de Google+.

(RF4.III) Descargar los datos de invitados de Gmail.

RF5. Feed del evento en la página del usuario

Se permitirá al usuario con rol de pareja escribir automáticamente en su muro de Facebook las últimas actualizaciones de su evento.

(RF5.I) Localización:

Se escribirá en el muro en el caso en el que la pareja lo permita en las preferencias, en las siguientes acciones:

- Subir una foto al evento.
- Realizar cualquier comentario.
- Añadir un nuevo post al evento.
- Editar un post existente.

(RF5.II) Sintaxis.

Se crearán entidades de Facebook de tipo “objeto”, “acción” e “historia” para poder relatar un evento. El resultado será el de compartir un comentario en el feed de Facebook del usuario cuyo título cuente con la siguiente sintaxis: “{Sujeto} {Acción} {Objeto} en {Web}”

(RF5.III) Tiempos verbales, género y número del sujeto.

Las historias generadas tendrán en cuenta los casos en los que hay más de un usuario que realiza la acción (una pareja), o el usuario sea anónimo. También se tendrán en cuenta varios tiempos verbales diferentes.

(RF5.IV) Enlace a la web.

Dichos mensajes contendrán en el apartado de descripción un link de la sección del evento que ha sido actualizada. En caso de hacer una referencia al evento en general, deberá incluir la referencia canónica del evento. La URL no podrá contener en ningún caso parámetros o información de sesión.

(RF5.V) Imagen.

Se deberá siempre utilizar una imagen. En el caso de subir una fotografía se mostrará la misma; en cualquier otro caso se utilizará o bien una imagen referente a la sección modificada, o la imagen de bienvenida del propio evento. En última instancia se utilizará el logotipo de Zankyou, con un tamaño no mayor a 600x315 píxeles.

(RF5. VI) Preferencias

Se añadirá a las preferencias de la aplicación la posibilidad de habilitar o deshabilitar de forma completa o parcial dicha funcionalidad, permitiendo que el usuario pueda elegir los tipos de acción en los que quiere generar este mensaje.

(RF5.VII) Aprobación por Facebook.

Para permitir este tipo de “Historias Personalizadas”, Facebook realiza una revisión de cada una de ellas. Ninguna historia puede ser publicada sin dicha aprobación, por lo que los objetos creados deberán seguir los criterios de validación de Facebook (disponibles en la sección “**Open Graph Submission Process**” de su manual [9]).

RF6. Mostrar datos de usuarios

(RF6.I) Mostrar avatar de un invitado en la pantalla de administración de la pareja.

Mostrará en la pantalla de administración de invitados de la pareja el avatar de los invitados, con la siguiente preferencia: Mostrar una imagen subida manualmente, en su ausencia el avatar que proporciona Facebook, un avatar obtenido de Gravatar o por último, una imagen por defecto.

(RF6. II) Permitir a la pareja subir una imagen de invitado.

Dicha imagen únicamente será visible por la pareja con motivos de administración, y será la que se muestre en el punto anterior. Debe permitir subir una foto guardada localmente y recortarla para un mejor resultado.

(RF6.III) Mostrar avatar de pareja.

Seguirá la misma preferencia del punto **RF6.I** y será visible tanto por la pareja como los invitados. Será el avatar que se muestre en cada participación de la pareja en su evento.

(RF6.IV) Permitir a un invitado subir su propio avatar al sistema.

De la misma forma que el punto **RF6.II**. Esta vez será el invitado quien pueda subir su propio avatar, que será el utilizado en cualquier participación de dicho usuario en el evento.

(RF6.V) Mostrar avatar de invitado.

Dicho avatar se escogerá igual que en el punto **RF6.I** y se mostrará en el perfil del invitado y en cada una de sus participaciones en el evento al que participa.

(RF6.VI) Mostrar clasificación de invitados.

Se mostrará una clasificación de invitados por número de visitar a la web. El diseño y la maqueta se la vista será proporcionada por Zankyou, y el trabajo consistirá en la integración de dicha maqueta al sistema.

(RF6. VII) Identificación automática de invitados dentro del evento.

Se identificará de forma automática al usuario en todas las acciones en las que requiera de identificación, como hacer un comentario en cualquier sección del evento, subir una foto, o postear en el foro o el blog del mismo.

RF7. Galería de fotos

(RF7.I) Subir fotos de Zankyou a la red social Facebook automáticamente.

Permitirá a la pareja subir los álbumes de fotos del evento a Facebook de forma automática. Se podrá desactivar dicha opción en la pantalla de administración.

(RF7.II) Subir un álbum de fotos de Facebook al evento.

A la hora de subir una foto, se permitirá a usuario con rol de pareja subir un álbum completo de fotos de Facebook. Se abrirá un formulario que permitirá seleccionar que álbum o álbumes desea subir y se bloqueará la página para evitar que se cierre antes de terminar la acción.

3.2 Requisitos no funcionales

REQUISITO	DESCRIPCIÓN
RNF1	Requisitos de usabilidad
RNF1.i	El <i>log-in</i> en la página web ha de poderse realizar rellenando como máximo un formulario, ya sea mediante Facebook o no. Se otorgará la opción de poder efectuar dicho <i>log-in</i> de manera automática al conectarse mediante su cuenta de Facebook. [10]
RNF1.ii	No se debe pedir en ningún momento más permisos de la API de Facebook de los que se vayan a utilizar. Además, todo permiso necesario para el funcionamiento de la aplicación se pedirá exclusivamente cuando sea estrictamente necesario. De esta forma el usuario podrá saber de una forma intuitiva para qué se pide dicho permiso. [10]
RNF1.iii	Los permisos de Facebook serán no bloqueantes: Si un usuario no acepta dar un permiso a la aplicación <u>no</u> se deberá volver a pedir de forma continuada hasta que acepte. La aplicación debe funcionar de manera normal exceptuando la funcionalidad que requiera de dicho permiso. [10]
RNF1.iv	Para facilitar y clarificar al usuario la interacción con Facebook, los botones que prestan estas nuevas funcionalidades deben seguir el patrón proporcionado por Facebook. [10]
RNF1.v	Se debe informar al usuario a la hora de conectarse con Facebook de que la aplicación no escribirá nada en su lugar sin su permiso. [10]
RNF1.vi	No se podrá pedir permisos de lectura de datos y escritura en Facebook al mismo tiempo, ya que dichos permisos se obtienen mediante dos peticiones diferentes y el usuario debería aceptar dos veces seguidas dichas peticiones. [10]

REQUISITO	DESCRIPCIÓN
RNF2	Requisitos de Disponibilidad
RNF2.i	El sistema debe funcionar correctamente en cualquier navegador, especialmente en las últimas versiones de Internet Explorer, Mozilla Firefox y Google Chrome.
RNF2.ii	La disponibilidad del servicio web no puede verse afectada por la disponibilidad del módulo. Si el módulo no se encuentra disponible, el servicio no puede interrumpirse. [10]
RNF2.iii	La disponibilidad del módulo debe ser igual a la disponibilidad de la API proporcionada por la red social Facebook.
RNF3	Requisitos de Seguridad
RNF3.i	El sistema debe cumplir las disposiciones recogidas en la Ley Orgánica de Datos Personales.
RNF4	Requisitos de Mantenibilidad
RNF4.i	El módulo debe cumplir con estándares de codificación de la empresa Zankyou Ventures para facilitar su mantenibilidad. [11]
RNF5	Requisitos de interfaz
RNF5.i	El módulo mantendrá el estilo y estructura presente en la página web.
RNF6	Requisitos de costo
RNF6.i	El sistema no deberá utilizar herramientas de pago o de terceros si no es estrictamente imprescindible.

Tabla 1: Requisitos no funcionales

4 IMPLEMENTACIÓN

4.1 Diseño e implementación

4.1.1 Identificación de usuarios (RF1)

Identificación de la pareja mediante Facebook

Implementación:

Antes de llevar a cabo la implementación existía en la aplicación un método para identificar al usuario mediante Facebook. Dicho método estaba disponible únicamente en parte de la aplicación y su uso era limitado, ya que servía exclusivamente para mantener los datos de sesión de un usuario.

Durante la implementación de este requisito funcional se ha movido dicho método a una zona accesible por toda la aplicación y mejorado para llevar a cabo las peticiones de permisos de Facebook al mismo tiempo que mantiene los datos de sesión. Por otro lado, ahora también se guardan los datos del usuario, como el identificador de Facebook para hacer futuras llamadas a la *Graph API*. [12]

En su implementación, dicho método mantiene una doble sesión (utilizando la cuenta de ambos servicios), y lleva a cabo acciones como crear o validar usuarios, ya que al validar el propio Facebook a sus usuarios, éstos son considerados válidos para nuestra aplicación.

A continuación se muestra el código referente a la identificación de usuario que requiere únicamente el email del usuario. El código utilizado para obtener otros permisos de Facebook se verá con más detenimiento en el siguiente apartado.

```
FB.getLoginStatus(function(response) {
  // Logged in and connected user, someone you know
  if (response.status === 'connected') {
    FB.api('/me', function(response) {
      callback(response);
    });
  }
  // No user session available, someone you dont know
} else {
  FB.login(function(response) {
    // User successfully logged in
    if (response.authResponse) {
      FB.api('/me', function(response) {
        callback(response);
      });
    }
    // User cancelled login
  } else {
    callback(false);
  }
  // Requests permissions for receiving the email
}, { scope : 'email' });
});
```

Tratamiento de permisos de Facebook

Implementación:

Facebook requiere de una serie de **permisos** para escribir en la página de Facebook de un usuario, u obtener datos del mismo. Esto es debido a que es necesario que el usuario conozca y establezca los límites de actuación de una aplicación externa a la hora de utilizar su cuenta de Facebook.

En un principio, cualquier usuario que entra por primera vez en una aplicación utilizando su cuenta de Facebook tendrá la posibilidad de otorgar permisos de lectura de sus campos públicos, como nombre o perfil, y su lista de amigos. Dichos permisos se piden de forma general cada vez que un usuario se conecta mediante Facebook a una APP externa.

Si la aplicación necesita cualquier otro permiso que no esté incluido dentro de los anteriores, ha de pedirlo expresamente. Para ello, se adjuntan los permisos necesarios al código destinado a conectarse con Facebook.

Para facilitar y generalizar su uso, se ha creado un método Javascript de conexión con Facebook accesible desde toda la aplicación mediante el cual se pueden pedir los permisos necesarios sin necesidad de modificar la estructura interna ni repetir código. Dicho método es activado por los elementos HTML y lee el atributo "**data-token**" para obtener los permisos que requiere la acción. En caso de no existir dicho atributo, no se pide ningún permiso adicional.

Todas las acciones de Facebook requieren que el usuario se haya identificado con Facebook, y haya otorgado los permisos necesarios. Por lo tanto, cada petición a la API de Facebook está precedida de una llamada al método anteriormente mencionado. El código de dicho método es el siguiente:

```
FB.api('/' + id + '/permissions', function (response) {
    // The user is not logged on Fb
    if (!response || response.error) {
        return false;
    }
    // The user actually have the token
    if (response.data[0][token] == 1) {
        callback(true);
    } else {
        // FB.Login method is used to request tokens.
        FB.login(function(response) {
            FB.api('/' + id + '/permissions', function (response) {
                // Fb has given us the token
                if (response.data[0][token] == 1) {
                    callback(true);
                    return true;
                }
            });
        }, {scope : token});
    }
});
```

Enlazar una cuenta de Facebook con una de Zankyou

Diseño:

Se ha permitido enlazar una cuenta de Facebook a la cuenta de un usuario registrado, utilizando su propio panel de administración **[Figura 8]**. El único requisito es que dicha cuenta de Facebook no puede estar ligada a otra cuenta de la aplicación. El enlace para ello se encuentra en el panel de administración del usuario. En caso de haber ligado dicho usuario ambas cuentas, se mostrará su imagen de perfil de Facebook. **[Figura 9]**



Figura 8: Enlazar una cuenta con Facebook



Figura 9: Cuenta enlazada con Facebook

Implementación:

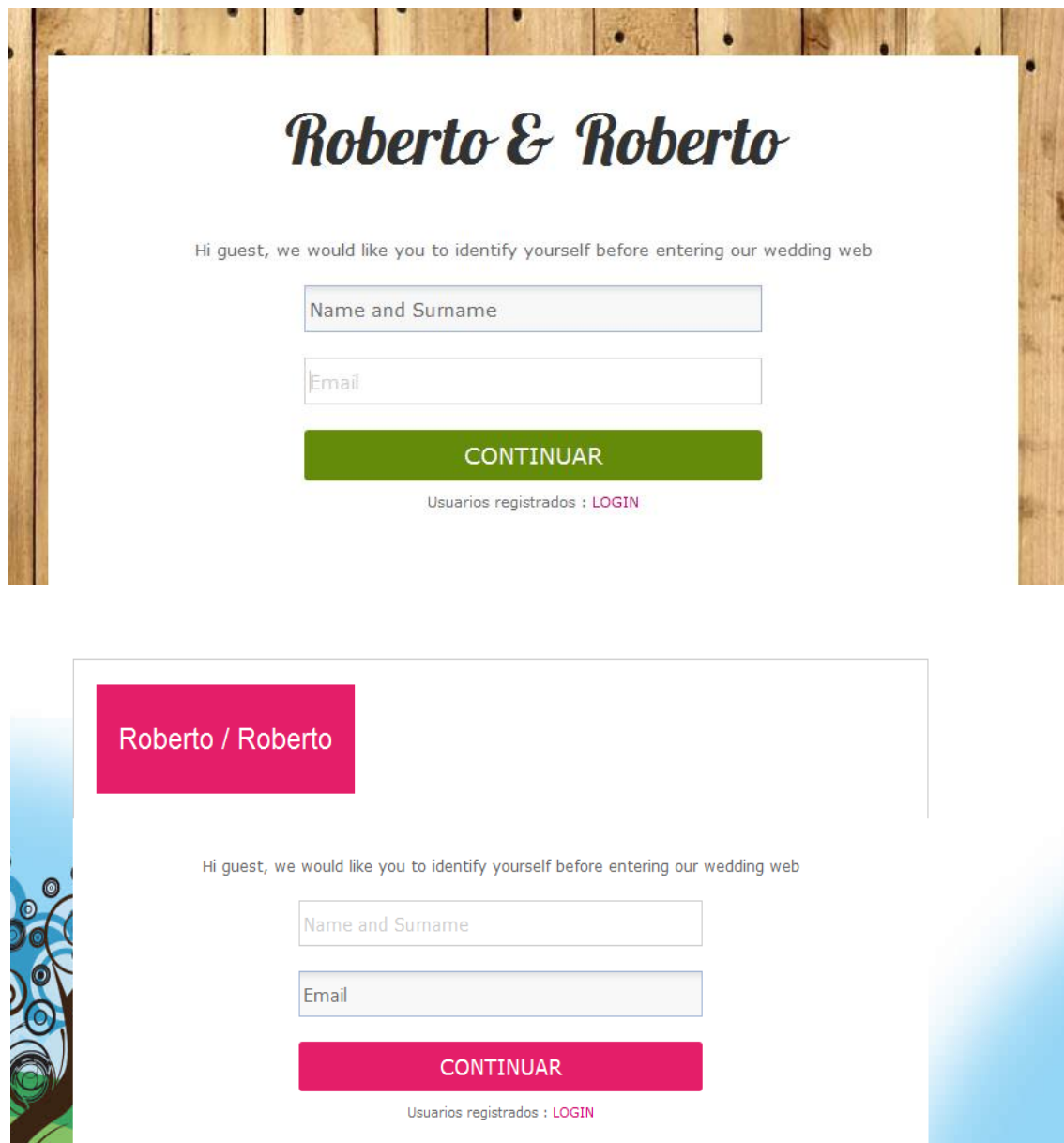
La acción de eliminar el enlace entre la cuentas de usuario de Zankyou y Facebook simplemente elimina la asociación entre ambas en la base de datos de Zankyou. En ningún caso se lleva a cabo en la aplicación un borrado de permisos de Facebook del usuario ni desconecta la cuenta de Facebook del mismo. Este comportamiento está destinado a minimizar el número de peticiones de Facebook y a asegurar un funcionamiento que no sea extraño para el usuario, ya que no es usual que una aplicación externa desconecta a un usuario de su cuenta de Facebook.

4.1.2 Identificación de invitados (RF2)

Diseño:

Se ha creado un formulario de identificación para el invitado y dispuesto en una nueva vista que formará la página principal de los eventos que requieran identificación. Cualquier usuario no registrado como invitado será redirigido a la página si intenta acceder a cualquier sección que requiera de identificación. Dicha vista se adapta a los colores y diseño del evento al que pertenece [Figura 10] y muestra el título del evento. Tanto diseño como maquetación han sido proporcionados por la empresa Zankyou Ventures.

A continuación se muestra el formulario de entrada en sus diferentes estados: [Figuras 10-12]



The figure displays two versions of a guest identification form for an event titled "Roberto & Roberto".

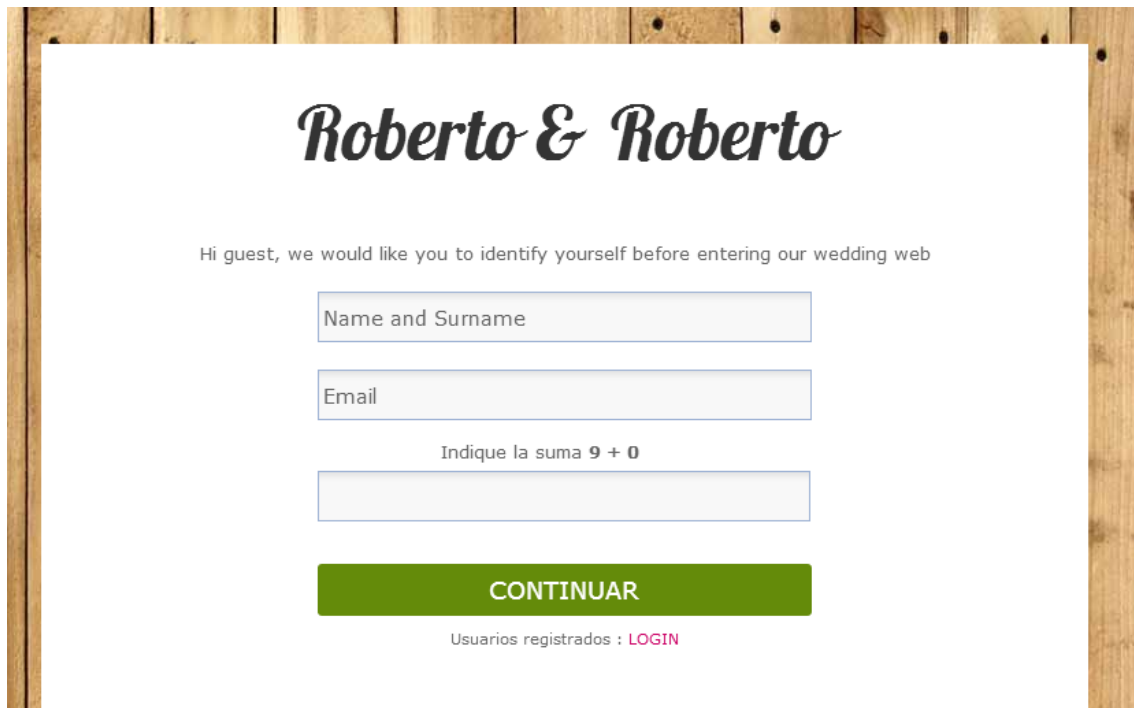
Top Version (Wood Background):

- Title: *Roberto & Roberto*
- Greeting: "Hi guest, we would like you to identify yourself before entering our wedding web"
- Input fields: "Name and Surname" and "Email"
- Button: "CONTINUAR" (green)
- Footer: "Usuarios registrados : LOGIN"

Bottom Version (Blue/White Background):

- Title: Roberto / Roberto
- Greeting: "Hi guest, we would like you to identify yourself before entering our wedding web"
- Input fields: "Name and Surname" and "Email"
- Button: "CONTINUAR" (pink)
- Footer: "Usuarios registrados : LOGIN"

Figura 10: Paneles de identificación de invitados



Roberto & Roberto

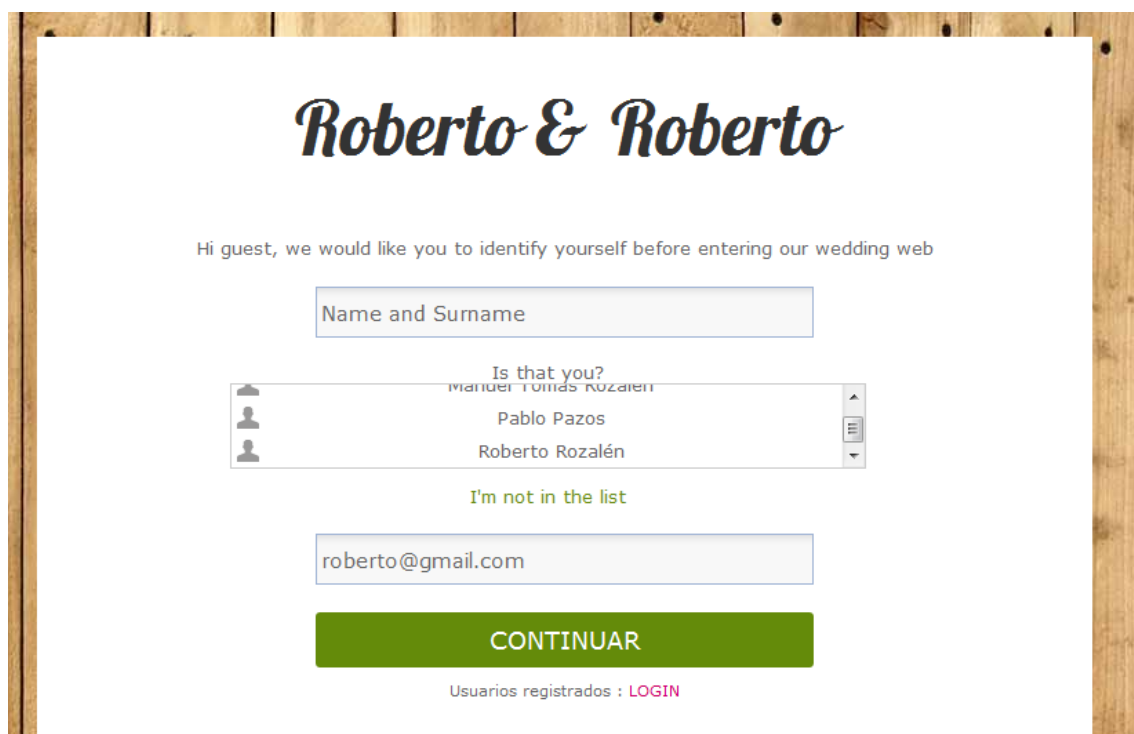
Hi guest, we would like you to identify yourself before entering our wedding web

Indique la suma **9 + 0**

CONTINUAR

Usuarios registrados : [LOGIN](#)







Figura 11: Panel de identificación de invitados (versión sin Javascript)



Roberto & Roberto

Hi guest, we would like you to identify yourself before entering our wedding web

Is that you?

	mander tomas rozalen	
	Pablo Pazos	
	Roberto Rozalén	

I'm not in the list

CONTINUAR

Usuarios registrados : [LOGIN](#)

Figura 12: Javascript de comparación de usuarios

Implementación:

El formulario muestra unos campos de texto con el nombre de la etiqueta en su interior. Dicha etiqueta se modifica mediante Javascript ante cualquier evento de click. En la versión sin Javascript dichas etiquetas se mantienen en el interior del campo de texto, por lo que el usuario debe borrarlas antes de completar el formulario.

Al mismo tiempo, el campo de email busca los contactos del evento mediante AJAX y los compara con el campo dado. En el caso de haber una coincidencia se ofrecen todos los nombres relacionados con dicho email.

Se permiten emails vacíos y varios usuarios con un mismo email. Para solventar los problemas que ello conlleva, se ha establecido una segunda clave primaria compuesta por nombre, apellido e email del usuario. Los datos entrantes del formulario son filtrados antes de establecer dicha clave primaria.

Al enviar el formulario sin errores, se guarda el contacto en la base de datos (en caso de no existir aún) y se almacenan los datos de sesión en un objeto de la siguiente estructura:

```
array($codeEvent => $idContact)
```

En el caso de intentar acceder al evento haciendo uso de un usuario propietario del mismo (integrantes de la pareja del evento), se lleva a cabo la identificación de los mismos de forma automática y no se permite al usuario utilizar este formulario.

4.1.3 Barra de invitado (RF3)

Diseño:

La barra de herramientas del invitado se adapta a los colores y diseño del evento al que pertenece, al igual que el requisito funcional anterior. A su vez, se redimensiona junto con el cliente que contenga la web, hasta cambiar completamente de forma para poder ser utilizada también en la versión para móviles.

A continuación podemos ver a continuación la integración completa de la barra teniendo en cuenta el estilo de la web [Figura 13].



Figura 13: Evento con barra de invitado

A continuación se muestran algunos ejemplos de diseño de dicha barra, en los que se tienen en cuenta algunos posibles casos. [Figura 14] La maquetación ha sido dada por la empresa Zankyou.

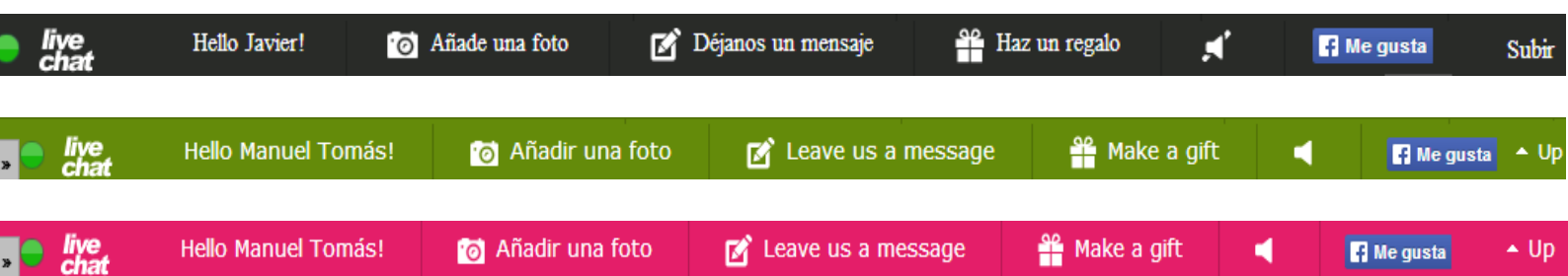


Figura 14: Diseños de barra de invitado

También podemos ver el aspecto de la versión para móviles de la barra [Figura 15], que cambia cuando el marco en el que se encuentra es de un ancho menor a un valor dado.



Figura 15: Versión móvil de la barra

Implementación:

El chat se ha implementado como un bloque para poder ser utilizado en varias vistas diferentes. Se ha añadido en cada *layout* del evento para ser mostrado en todos ellos.

Dicho bloque se carga en cada evento siempre y cuando la pareja haya habilitado la barra en el panel de administración. Es el propio bloque el que determina las secciones que debe mostrar, o si debe ocultar la barra completa.

El bloque toma el nombre del usuario de la sesión para mostrarlo en la barra, o muestra un enlace a la página de identificación en caso de no encontrarlo.

Ya que la barra no se muestra en todas las secciones del evento, se ha creado un **dobles filtro** utilizando **lista blanca y lista negra** para definir las acciones en las que se mostrará dicha barra. El filtro por lista blanca permite especificar controlador y acciones asociadas al mismo. En caso de no contener acciones, se toman como válidas todas las acciones contenidas dentro del controlador. Una vez filtrada la lista blanca, se le aplica al conjunto resultante una lista negra que recibe pares controlador/acción para bloquear determinadas acciones.

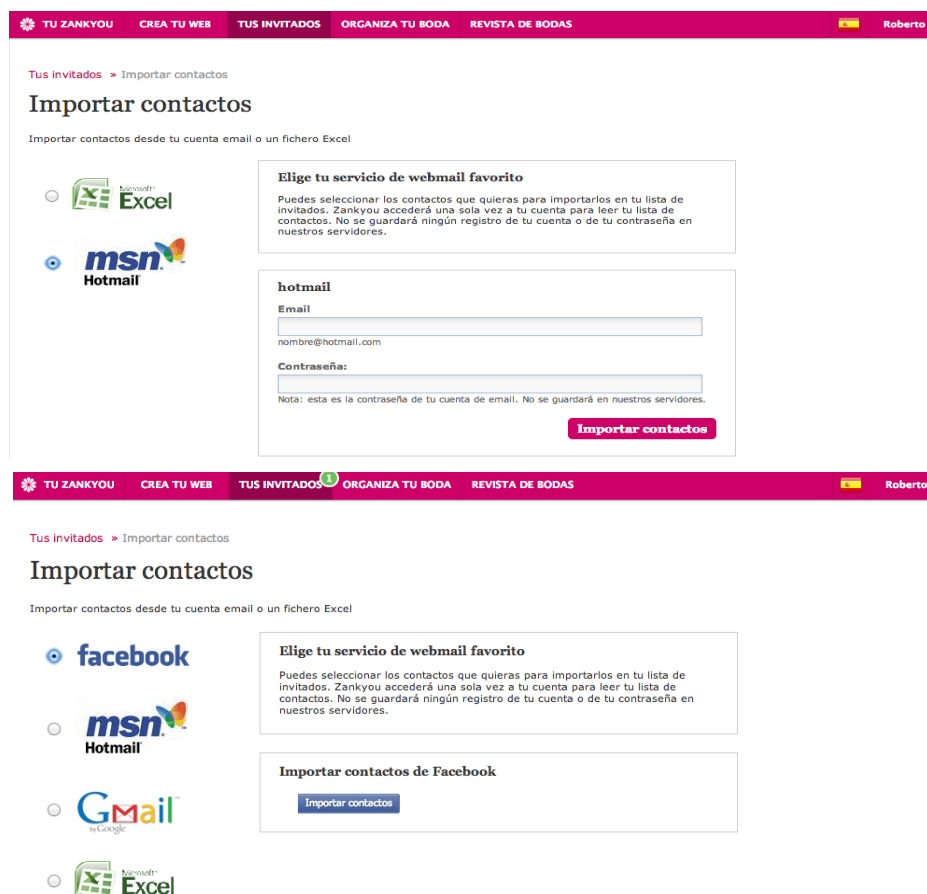
El bloque tiene asociado también un fichero Javascript encargado de modificar el icono del botón de silenciar o activar sonido, y mover el *scroll* de la página al inicio de forma progresiva mediante un enlace situado en la parte derecha de la barra.

4.1.4 Adición de invitados desde otros servicios (RF4)

Diseño:

Se ha añadido la funcionalidad de añadir usuarios a través de una cuenta de Facebook y una cuenta de correo de Gmail. El diseño de dichas funcionalidades ha seguido el mismo de las funcionalidades ya presentes (añadir usuario a través de Excel y Hotmail).

A continuación se muestra una comparación entre el estado de las vistas antes de añadir las nuevas funcionalidades y el resultado final [Figuras 16 y 17]:



The figure displays two screenshots of a web application interface for importing contacts. The top screenshot shows the 'Importar contactos' page with options for Excel, msn, and Hotmail. The bottom screenshot shows the same page with additional options for Facebook and Gmail.

Top Screenshot (Before):

- Navigation bar: TU ZANKYOU, CREA TU WEB, TUS INVITADOS, ORGANIZA TU BODA, REVISTA DE BODAS. User: Roberto.
- Breadcrumb: Tus invitados » Importar contactos
- Section: Importar contactos
- Text: Importar contactos desde tu cuenta email o un fichero Excel
- Options:
 - Excel (selected)
 - msn
 - Hotmail
- Form:
 - Elige tu servicio de webmail favorito
 - Puedes seleccionar los contactos que quieras para importarlos en tu lista de invitados. Zankyou accederá una sola vez a tu cuenta para leer tu lista de contactos. No se guardará ningún registro de tu cuenta o de tu contraseña en nuestros servidores.
 - hotmail
 - Email: nombre@hotmail.com
 - Contraseña:
 - Nota: esta es la contraseña de tu cuenta de email. No se guardará en nuestros servidores.
 - Importar contactos

Bottom Screenshot (After):

- Navigation bar: TU ZANKYOU, CREA TU WEB, TUS INVITADOS, ORGANIZA TU BODA, REVISTA DE BODAS. User: Roberto.
- Breadcrumb: Tus invitados » Importar contactos
- Section: Importar contactos
- Text: Importar contactos desde tu cuenta email o un fichero Excel
- Options:
 - facebook (selected)
 - msn
 - Hotmail
 - Gmail
 - Excel
- Form:
 - Elige tu servicio de webmail favorito
 - Puedes seleccionar los contactos que quieras para importarlos en tu lista de invitados. Zankyou accederá una sola vez a tu cuenta para leer tu lista de contactos. No se guardará ningún registro de tu cuenta o de tu contraseña en nuestros servidores.
 - Importar contactos de Facebook
 - Importar contactos

Figura 16: Comparativa de importación de contactos



The figure displays two versions of the 'Importar contactos' block. The left version shows the block with a reminder to import contacts in bulk. The right version shows the block with additional options for Facebook and Gmail.

Left Version (Before):

- Section: Importar contactos
- Text: Recuerde que también puede importar contactos de forma masiva
- Options:
 - Excel
 - msn
 - Hotmail

Right Version (After):

- Section: Importar contactos
- Text: Recuerde que también puede importar contactos de forma masiva
- Options:
 - Excel
 - msn
 - Hotmail
 - facebook
 - Gmail

Figura 17: Comparativa del bloque de importación de contactos

Implementación:

La sección destinada a la importación de contactos mediante Gmail utiliza la misma lógica que la usada mediante Hotmail. Se realiza una única llamada a la API de Gmail indicando correo y contraseña, y se mandan los contactos recibidos a otra acción mediante un método POST.

Para la sección de Facebook se ha añadido una sección cuyo único contenido es un botón activado por Javascript. Dicho botón utiliza el método para identificarse con Facebook mencionado anteriormente en el punto 4.1.1 y a continuación hace una llamada a la API de Facebook para pedir los datos de los amigos del usuario. Dichos datos se filtran y se mandan utilizando Javascript a la siguiente acción mediante un método POST, donde son utilizados para rellenar el formulario que vemos a continuación [Figura 18]. Se guarda, entre otros, el identificador de Facebook de dichos usuarios para poder obtener los avatares de los mismos.



Figura 18: Formulario de contactos

El código utilizado para obtener los amigos Facebook de un usuario utilizando la Graph API es el siguiente. Como se puede observar hay que indicar a la API el identificador del usuario (siendo "me" el usuario actualmente conectado) y el nodo a pedir (en este caso, "friends"):

```
FB.api("/me/friends", function (friends) {
    if (!friends || friends.error) return false;
    // Send data to webmailContactsAction
    _sendPostToUrl(sVars.webmail_contacts_url, friends.data);
});
```

La integración de invitados con Google+ no ha sido implementada debido a que la complejidad que conlleva es mayor que la funcionalidad que aporta. Ver secciones [4.2 Validación] y [5 Conclusiones y líneas futuras].

4.1.5 Publicación en Facebook (RF5)

La herramienta Graph API de Facebook permite no sólo compartir un comentario escrito por el usuario o la aplicación, sino crear un objeto completamente personalizable muy superior a un comentario común. A continuación [Figuras 19 y 20] podemos comprobar la diferencia entre ambos objetos.



Figura 19: Facebook Share

Custom Stories

Post rich stories from your app using a strongly typed API.

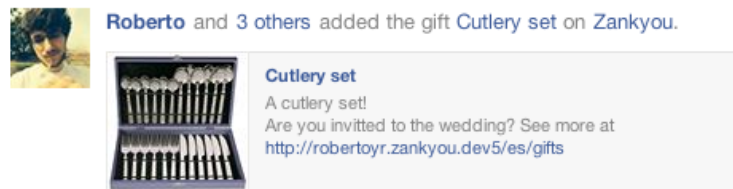


Figura 20: Facebook Story

Como podemos ver, ambos objetos se componen de un **título** y un **bloque de contenido**, sin embargo se pueden apreciar muchas diferencias entre ambos. A continuación se detalla la estructura de ambos objetos:

Título:

El título de la función de **compartir** es genérico para todo artículo. [13] Se compone del **nombre del usuario** y un **enlace a la página** compartida. Sin embargo, el título de la **historia personalizada** sigue la siguiente sintaxis: [14]

{Sujeto} {Acción} {Objeto} en {Web}

- El **sujeto** puede ser el nombre del usuario o estar compuesto por varios usuarios.
- La **acción** es otro objeto personalizable de Facebook. Corresponde a un verbo y puede contemplar todos los tiempos verbales posibles.
- El **objeto** incluye el artículo con el que queremos utilizarlo, ya sea determinado o indeterminado, y tiene en cuenta el género y número del mismo.
- La **web** corresponde al enlace de la página a compartir como en el caso de la función *share*.

Contenido:

En ambos casos la estructura es la misma. Está compuesto por los siguientes elementos:

- **Imagen:** En caso de tener un tamaño mayor a 600x315 píxeles el contenido se mostrará en un formato mayor, con la imagen como elemento principal.
- **Título:** Contiene un enlace al contenido compartido.
- **Subtítulo:** Breve descripción del artículo en una línea.
- **Descripción:** Descripción del contenido de dos o tres líneas de extensión.

Una vez explicado el contenido, es necesario explicar cómo creamos este nuevo elemento. Para crear una historia debemos crear primero los elementos que la componen. Dichos elementos son **acciones** y **objetos**. En nuestro caso hemos creado los siguientes elementos:

[Figuras 21 y 22]

Historias		Tipos de acción	Object Types
Object Types	Properties		
Article	Published Time, Modified Time, Expiration Time, Au...		Obtener código · ×
Event			Obtener código · ×
Gift			Obtener código · ×
Hotel			Obtener código · ×
Photo			Obtener código · ×
Section			Obtener código · ×

Figura 21: Objetos personalizados

Historias		Tipos de acción	Object Types
Tipos de acción	Associated Objects		
● Add	Section, Gift, Hotel, Event		Obtener código · ×
● Comment	Section		Obtener código · ×
● Like	Object		Obtener código · ×
● Update	Section, Gift, Event		Obtener código · ×
● Upload	Photo		Obtener código · ×

Figura 22: Acciones personalizadas

Finalmente, podemos **combinar** unas **acciones y objetos** creando historias independientes [Figura 23], que publicaremos de forma automática utilizando la versión HTML utilizando un método POST desde el código PHP, o utilizando la versión para Javascript de la llamada a la API en caso de ser necesario.

Historias	Tipos de acción	Object Types
● Add a Gift		Vista Previa · Obtener código · x
● Add a Hotel		Vista Previa · Obtener código · x
● Add a Section		Vista Previa · Obtener código · x
● Add an Event		Vista Previa · Obtener código · x
● Comment a Section		Vista Previa · Obtener código · x
● Update a Gift		Vista Previa · Obtener código · x
● Update a Section		Vista Previa · Obtener código · x
● Update an Event		Vista Previa · Obtener código · x
● Upload a Photo		Vista Previa · Obtener código · x

Figura 23: Historias personalizadas

Para ello, debemos crear dichos elementos llamando a la aplicación utilizando el código proporcionado por Facebook. Dicha llamada está sigue la siguiente sintaxis:

***{URL} / {ID del usuario} / {Nombre de la aplicación} : {Nombre del elemento} ?
access_token=ACCESS_TOKEN & method=POST & {Identificación del elemento}***

De tal forma, si queremos crear la historia de "Añadir un hotel" debemos crear el objeto "Hotel" con el código expuesto a continuación:

```
https://graph.facebook.com/me/objects/zankyot:hotel?

access_token=ACCESS_TOKEN&

method=POST&

object=%7B%22app_id%22%3A290267185975%2C%22type%22%3A%22%5C%22zankyot%3Ahotel%5C%22%2C%22url%22%3A%22%5C%22http%3A%5C%2F%5C%2Fsamples.ogp.me%5C%2F10151961154740976%5C%22%2C%22title%22%3A%22%5C%22Sample+Hotel%5C%22%2C%22image%22%3A%22%5C%22https%3A%5C%2F%5C%2Ffbstatic-
```

```
a.akamaihd.net%5C%2Fimages%5C%2Fdevsite%5C%2Fattachment_blank.png%5C%22%22%2C%22description%22%3A%22%5C%22%5C%22%22%7D
```

Y por último, debemos crear la acción asociada al objeto anterior, donde el apartado "hotel" es la referencia al elemento anteriormente creado:

```
https://graph.facebook.com/me/zankyou:add?  
  
access_token=ACCESS_TOKEN&  
  
method=POST&  
  
hotel=http%3A%2F%2Fsamples.ogp.me%2F10151961152195976
```

Una vez creados los objetos se han añadido a las acciones de la web que los activan. Por ejemplo, la historia "Añadir un regalo" se lanza cuando la pareja envía el formulario destinado a añadir un regalo al evento. Si la acción asociada retorna un error, no se efectúa la llamada a la API. También se ha añadido la opción de privacidad de dicha funcionalidad, mediante la cual los usuarios pueden activar o desactivar la posibilidad de compartir automáticamente con Facebook las acciones que llevan a cabo en Zankyou.

La última fase es la fase de **validación de Facebook [9]**. Para llevar a cabo esta fase hay que asegurarse de haber completado las anteriores sin errores y tener la funcionalidad que se requiera ya implementada. En dicha fase, se le manda una **petición a Facebook** para que revise los elementos creados (historias, acciones y objetos) y de por buenos tanto dichos elementos como su uso dentro de la web. Para ello, se adjunta el flujo dentro de la web mediante el cual se disparan las acciones de Facebook que queremos comprobar. Esta fase es necesaria ya que Facebook no permite publicar ninguna historia personalizada antes de haberla comprobado para asegurar el buen uso de la misma.

Dicha fase no ha sido implementada aún, ya que las acciones implementadas están sujetas a constantes cambios por el momento. Por ello, esta funcionalidad se encuentra únicamente en el servidor de desarrollo.

4.1.6 Datos de usuarios (RF6)

Subir y mostrar avatar de un invitado en la pantalla de administración de la pareja.

La pareja cuenta con un panel de administración para editar la información de los invitados. Se ha añadido la opción de **ver y modificar la imagen de avatar** de los invitados desde dicho panel de control [Figura 24]. Para modificar la imagen el usuario puede pulsar en el link "Cambiar avatar" o en la propia imagen y se abre mediante Javascript una ventana para subir imágenes desde el equipo local.



Figura 24: Avatar de invitado

Además, se ha añadido un método para recortar la imagen subida por el usuario. El usuario puede elegir la posición y tamaño de la sección de la imagen a mostrar y serán almacenadas en el sistema tanto la imagen original como dicha sección para permitir un acceso rápido al recurso. [Figura 25]

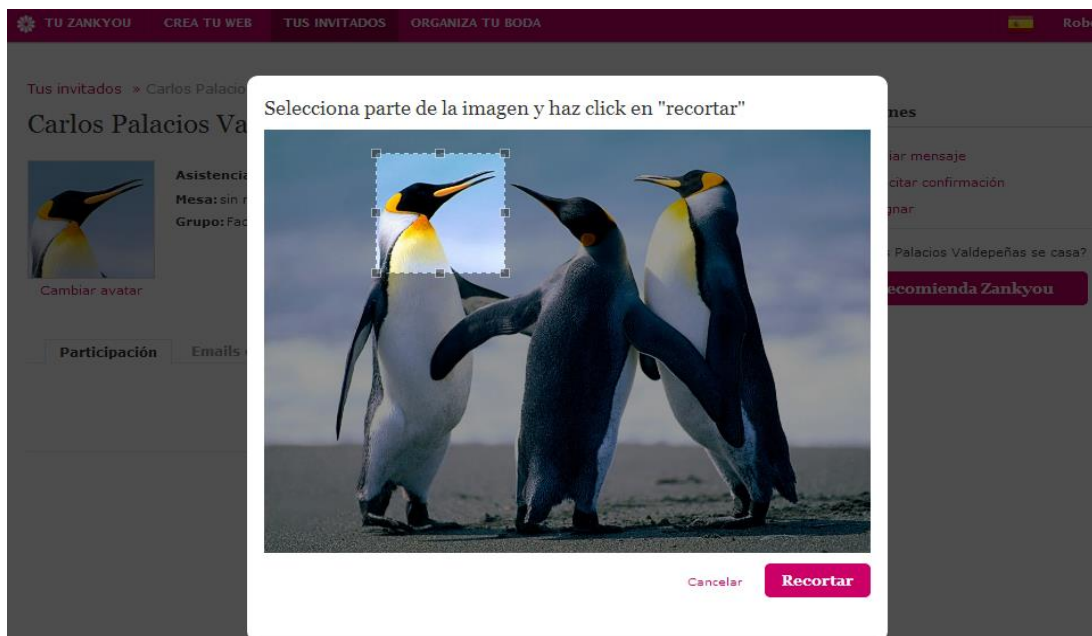


Figura 25: Sección de imagen

Mientras se elige la imagen a mostrar, el propio avatar muestra la sección elegida, que se modifica en tiempo real mediante Javascript mientras el usuario determina la sección a guardar.

Se mostrará dicha imagen recortada hasta el momento en el que el usuario pulse los botones de guardar o cancelar, momento en el que se subirá la imagen al sistema y se enlazarará con el avatar del usuario de forma permanente, o se desechará, respectivamente.

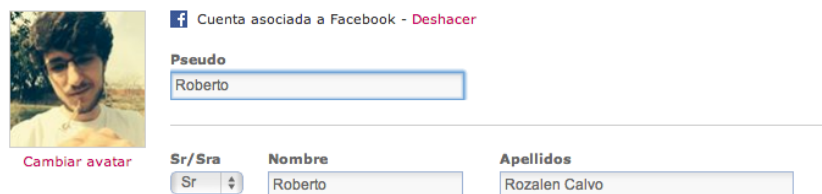
Mostrar avatar de pareja.

En caso de estar el usuario conectado con Facebook, su panel de administración muestra su **foto de perfil de Facebook [Figura 26]**, a no ser que el usuario haya subido manualmente una imagen al sistema, en cuyo caso se mostrará dicha imagen. **La imagen obtenida mediante Facebook no se almacena**, sino que se pide a la Graph API de Facebook en los casos en los que es necesario utilizando la siguiente llamada a la API mediante Javascript:

```
'http://graph.facebook.com/__FB_ID__/picture';
```

Donde FB_ID corresponde al identificador de Facebook del usuario, almacenado en nuestra aplicación. Dicho dato es público, por lo que no es necesario llevar a cabo ninguna petición de permisos especiales para obtener dicha imagen.

Datos del perfil



The screenshot shows a user profile form. At the top left is a profile picture of a man with curly hair. To its right is a Facebook icon and the text 'Cuenta asociada a Facebook - Deshacer'. Below the picture is a 'Cambiar avatar' link. The form contains the following fields:

- Pseudo:** A text input field containing 'Roberto'.
- Sr/Sra:** A dropdown menu with 'Sr' selected.
- Nombre:** A text input field containing 'Roberto'.
- Apellidos:** A text input field containing 'Rozalen Calvo'.

Figura 26: Avatar de usuario obtenido mediante Facebook

Incluye también la misma funcionalidad de **subida y recorte de imágenes** mencionada anteriormente.

Permitir a un invitado subir su propio avatar al sistema.

Se ha creado una nueva vista para el invitado desde la que puede editar sus datos y establecer la imagen de avatar. Dicha vista es análoga a la de edición de la pareja, con la diferencia de que el invitado tiene menos datos que mostrar o editar. Al igual que la anterior, esta vista también cuenta con la opción de ligar su cuenta o no a Facebook, lo que permite utilizar la imagen de perfil de Facebook del usuario como avatar de la aplicación.



Figura 27: Sección de administración del invitado

Mostrar avatar de invitado.

Se ha añadido la imagen de avatar en las acciones que llevan a cabo los usuarios, tanto novios como pareja. De esta forma se remarca la entidad del usuario que ha publicado una entrada en la lista de invitados, hecho un regalo, etc.

Al igual que en el resto de funcionalidades de este género, se ofrece la posibilidad de tomar la foto de perfil de Facebook del usuario, que será reemplazada si el usuario decide subir una imagen de avatar manualmente.

Mostrar clasificación de invitados.

No implementada. Ver secciones [4.2 Validación] y [5 Conclusiones y líneas futuras].

Identificación automática de invitados dentro del evento.

Antes de la implementación de la pantalla de identificación de invitado, cualquier participante en el evento era anónimo y tenía que introducir su nombre y email en cada acción en la que participase activamente, como postear en el foro o hacer comentarios en cualquier sección del evento [Figura 29].



Comentarios (1) [Escribir un comentario](#)

Título comentario

Comentar foto

Email

Nombre

Indique la suma 3 + 4

Cancelar **Guardar comentario**

Figura 28: Comentar una foto

Al desarrollar esta funcionalidad, mantenemos la posibilidad de tener usuarios anónimos que tengan que introducir sus datos a la hora de participar en el evento, pero facilitamos la participación de los usuarios identificados obteniendo directamente sus datos de la sesión.

A su vez se han unificado las partes comunes y el comportamiento por defecto de los formularios destinados a compartir un comentario en el evento. Dichos formularios heredan de uno común que además especifica la estructura de los mismos (utilizando un **patrón Template**). De esta forma además diferenciamos tres tipos de elementos para dichos formularios:

- **Elementos comunes:** Presentes en cada formulario en todo momento.
- **Elementos de usuario anónimo:** Presentes únicamente cuando no se encuentra información del participante en la sesión.
- **Elementos de usuario identificado:** Utilizados para rellenar de forma automática con datos del participante obtenidos de la sesión.

4.1.7 Galería de imágenes (RF7)

No implementada. Ver secciones [4.2 Validación] y [5 Conclusiones y líneas futuras].

4.1.8 Otras funcionalidades y problemas encontrados

Facebook view helpers

Se ha creado una serie de **métodos en código PHP** cuyo objetivo es **proporcionar datos comunes a las vistas**, que no tienen acceso a los datos del modelo de forma implícita. Dichos métodos están pensados para ser utilizados independientemente del contexto, por lo que actúan como métodos **estáticos** y únicamente están disponibles para las vistas.

El método principal es el **Model_View_Facebook** y contiene el identificador de la aplicación en Facebook (**APP_ID**) y su clave privada (**APP_SECRET**) para ser utilizadas en las llamadas a la Graph API de Facebook. De dicho método heredan todos los métodos relacionados con Facebook, de tal forma que no se repiten datos como la APP_ID. Esto permite además establecer la estructura común a modo de **patrón Template**, de modo que únicamente se modifique la URL de acceso a la API de los métodos hijos para modificar su comportamiento.

```
/**
 * ID token
 * Target field. This usually is a user id, but it can be anything
 */
const FB_ID_REPLACE_TOKEN = '__FB_ID__';

/**
 * Zankyou public and private tokens
 */
const FB_APP_ID_REPLACE_TOKEN = '__FB_APP_ID__';
const FB_APP_SECRET_REPLACE_TOKEN = '__FB_APP_SECRET__';

/**
 * Zankyou public and private Id
 */
const APP_SECRET = 'XXXXXXXXXXXXXXXXX';
const APP_ID = 'XXXXXXXXXXXXXXXXX';
```

De esa forma, cada helper únicamente necesita cambiar la URL de la petición a la API de Facebook. A continuación se muestran las clases implementadas con algunos ejemplos de la URL utilizada para la petición:

- **View_Helper_Facebook_Avatar:** Se encarga de pedir la foto de perfil de facebook y devolver un enlace a la misma del tamaño requerido. La llamada a la API utilizada es la siguiente:

```
const FB_PICTURE_URL = 'graph.facebook.com/__FB_ID__/picture';
```

- **View_Helper_Facebook_Like:** Encargado de obtener el número de *likes* que tiene una URL dada. La *query* utilizada para ello es la siguiente:

```
const FB_BASE = 'https://graph.facebook.com/fql?q=';  
const FB_QUERY = 'SELECT%20url,like_count%20FROM%20link_stat%20WHERE%20url=%27__FB_ID__%27';
```

- **View_Helper_Facebook_LikeButton:** Devuelve el código HTML de un botón de *like* de Facebook en su versión de *iFrame*. [15]
- **View_Helper_Dialog_Url:** Devuelve el código HTML de un botón para compartir un comentario en Facebook. Dicho comentario puede ser escrito por el usuario o por la aplicación de forma automática. La llamada requerida es la siguiente:

```
const URL_FEED = 'https://www.facebook.com/dialog/feed';
```

Nivel de publicación

En el momento en el que un usuario comparte contenido de forma activa con Facebook desde otra aplicación, se da a elegir al usuario el **nivel de publicación** que desea (**sólo yo | amigos | todos**). El usuario puede ver el nivel de publicación por defecto determinado por la aplicación y modificarlo para dicha acción.

En el caso de compartir contenido en Facebook de forma automática desde una aplicación, la aplicación debe pedir el permiso de Facebook **publish_actions** al usuario, y únicamente puede compartir si dicho usuario lo otorga. Durante este proceso el usuario determina también el nivel de privacidad que desea para dicha aplicación. Dicho nivel de privacidad únicamente puede ser modificado por el usuario en su panel de administración de Facebook o aceptando nuevamente el permiso **publish_actions**.

Desgraciadamente, existe un tercer caso que crea una situación de incertidumbre. Un usuario puede compartir de forma explícita contenido utilizando un botón de Like con la funcionalidad de compartir embebida. Cuando el usuario hace Like, se despliega un campo de texto para compartir un comentario acerca de la URL apuntada por el Like. Dicho campo no tiene nivel de privacidad. Sin embargo, escoge un nivel de privacidad, comparte el contenido introducido por el usuario y establece el nivel de privacidad del usuario para la aplicación. Actualmente escoge el nivel de privacidad determinado por la aplicación, pero en algún momento establecía el nivel de privacidad mínimo para cualquier usuario, por lo que ha generado el siguiente problema:

El **problema presentado** ha sido el siguiente: **Se estableció el nivel de privacidad para gran parte de los usuarios como "Sólo yo" de forma automática y sólo ellos pueden ver cualquier contenido que comparten.**

La **solución** es cambiar el nivel de privacidad que el usuario establece para la aplicación. Sin embargo, la aplicación no tiene permiso para hacer el cambio, sino que tiene que ser el propio usuario el que lo haga. Se han propuesto las siguientes soluciones para resolver este problema:

- Pedir un permiso de **publish actions** que no necesitamos, lo que puede minar la confianza del usuario hacia la aplicación y conllevar un gran rechazo.
- Informar de forma general a los usuarios de que comprueben su nivel de privacidad de la aplicación en Facebook y ofrecerles un enlace al panel de control de Facebook.
- No utilizar este tipo de comentario, lo cual conlleva utilizar funcionalidad de Facebook cuya respuesta es indeterminada, puesto que Facebook no contempla esa posibilidad. Además, requeriría un gran cambio en la aplicación, puesto que habría que modificar el aspecto externo de la página para mantener la funcionalidad de **compartir** sin utilizar el mismo botón de *like*.
- Cambiar el APP ID de la aplicación para reiniciar los datos que los usuarios tuvieran acerca de la aplicación. También conlleva la pérdida de todos los datos y estadísticas que tenemos de Facebook.

Tras un largo estudio del problema se ha determinado **mantener la aplicación actual y crear una nueva exclusivamente para las estadísticas de las funciones de "Me gusta" y "Compartir"**. De esta forma mantenemos los datos y estadísticas de Facebook y por otro lado desligamos las estadísticas de artículos que utilizamos de Facebook de aspectos funcionales como la identificación de usuarios o la petición de datos. En las páginas cuyo contenido pueda ser compartido de la forma anterior se ha establecido una **meta-cabecera indicando el identificador del administrador en lugar del APP_ID** de la aplicación, **desligando de esta forma dichos botones de nuestra aplicación**. La nueva APP mantendrá por defecto el nivel de privacidad en "Todos", eliminando de esa forma el problema.

4.2 Validación

Una vez acabada la fase de implementación, se han valorado los requisitos funcionales implementados. Recordemos que habíamos establecido 7 requisitos funcionales, con un número variable de apartados para cada requisito. A continuación se muestra la tabla de requisitos funcionales implementados [Tabla 2], donde las filas se asocian a requisitos y las columnas a apartados.

De esta forma, consideramos que un requisito funcional ha sido correctamente implementado si todos sus apartados han sido implementados.

Requisitos funcionales	I	II	III	IV	V	VI	VII
RF1	✓	✓	✓	✓			
RF2	✓	✓	✓	✓	✓	✓	
RF3	✓	✓	✓	✓	✓		
RF4	✓	✓	✗				
RF5	✓	✓	✓	✓	✓	✓	✗
RF6	✓	✓	✓	✓	✓	✗	✓
RF7	✗	✗					

Tabla 2: Validación de apartados de requisitos funcionales

✓: Requisitos funcionales implementados
✗: Requisitos funcionales no implementados

Como podemos observar, algunos apartados no han sido implementados. A continuación se explica la situación de cada apartado y la razón de la misma.

- **RF4.III. Descargar los datos de invitados de Gmail**

Se ha estimado que la carga en cuanto a código que supone esta funcionalidad supera los beneficios de la misma, por lo que no será implementada hasta el momento en el que se quieran llevar a cabo otro tipo de funcionalidades relacionadas con Gmail.

- **RF5.VII. Aprobación por Facebook**

Las historias personalizadas creadas durante este proyecto están siendo sometidas a constantes cambios de forma temporal debido a los cambios estructurales que ha sufrido la aplicación durante la reestructuración de los elemento Evento e Invitado. No

se enviará una petición de aprobación a Facebook hasta que se determinen de una forma las funcionalidades relacionadas con dichos elementos.

- **RF6.VI. Mostrar clasificación de invitados y RF7. Galería de fotos**

No ha sido posible implementar dichas funcionalidades por falta de tiempo. Ambos requisitos funcionales se plantearán como futuras mejoras en el **apartado 5** del documento.

4.3 Aplicación a un caso real

La funcionalidad implementada en el transcurso de este proyecto ha sido desarrollada y añadida a la página web de Zankyou Ventures **[8]**. Dicha web mantendrá las mejoras desarrolladas y planteará el futuro desarrollo de otras mejoras posibles propuestas en el **apartado 5. Conclusiones y líneas futuras** de este documento.

Dicha web se utiliza actualmente en **19 países** y alcanza las **120.000 visitas diarias**. Cuenta actualmente con **260.000 usuarios** y **150.000 eventos** diferentes.

Cabe decir que debido al número de usuarios el impacto que conlleva cada cambio en la web es muy considerable, por lo que se han introducido las mejoras desarrolladas de la forma más segura posible, ya que un fallo en diseño, implementación o funcionalidad de un nuevo requisito habría resultado ser fatal.

Aunque ha transcurrido un periodo de tiempo relativamente corto desde la implementación de las mejoras mencionadas anteriormente, se ha obtenido una respuesta por parte de los usuarios, que analizaremos a continuación en el punto **4.4 Evaluación de resultados**.

4.4 Evaluación de resultados

Gracias a las herramientas de *feedback* y estadísticas que proporciona Facebook se ha podido evaluar la respuesta que ha tenido el trabajo desarrollado en el documento.

Los requisitos funcionales relacionados con Facebook se han implementado en el transcurso de los meses **Febrero, Marzo y Abril**. A continuación podemos observar algunos datos interesantes [Figuras 30 a 33]:



Figura 29: Estadísticas generales de la aplicación

En el cuadro anterior podemos observar las estadísticas generales de la aplicación en el periodo de tiempo desde el **15 de Febrero al 11 de Mayo**.

Los porcentajes indicados en la figura anterior no deben tomarse como referencia, ya que son el resultado de la comparación de los datos del día 15/02/2014 con el 11/05/2014, por lo que no representan de forma fiable la evolución de dichos datos.

Los datos más significativos son el número de **referencias a la aplicación** desde Facebook, que ha ascendido hasta **15.2k** gracias a las funcionalidades implementadas dedicadas a compartir la aplicación con Facebook, tales como *like*, *share* o *custom story*, y a la publicidad realizada por la propia empresa.

Por otro lado podemos ver como se han producido **3.1k** nuevos inicios de sesión con Facebook en este periodo de tiempo, lo que es un buen indicativo de la relevancia de las tareas implementadas.

Ya que los datos anteriores no describen el comportamiento de los datos a lo largo del tiempo, a continuación vemos con más detalle la información de inicios de sesión [Figura 31]:

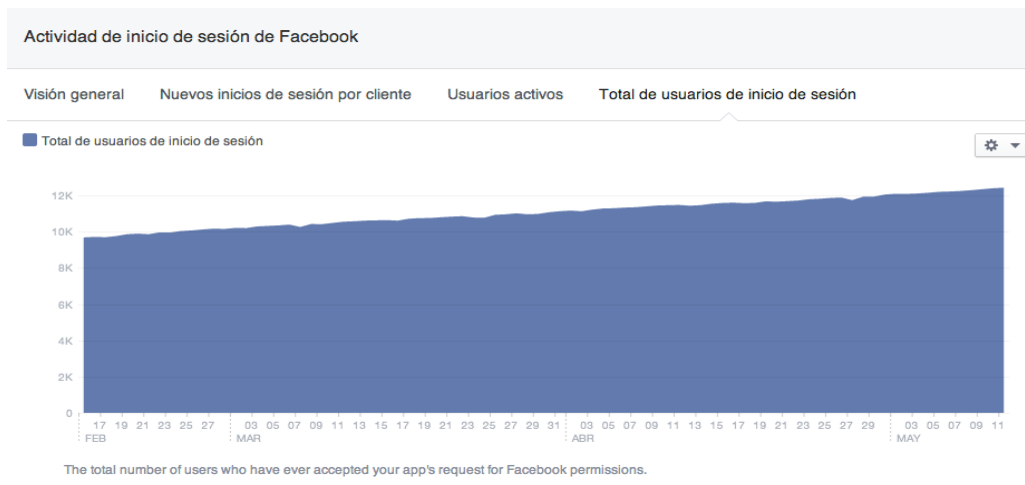


Figura 30: Número total de inicios de sesión

Como se puede observar, el número de inicios de sesión diarios en nuestra aplicación mediante Facebook ha aumentado de forma lineal. El **número de inicios de sesión diarios mediante Facebook** (que podría equipararse al número de usuarios que se conectan mediante Facebook a nuestra aplicación diariamente) ha aumentado de aproximadamente **10k usuarios en fecha 15 de Febrero** a **12k usuarios el 11 de Marzo**. Esto supone un **aumento de un 20%** en el número de usuarios que utilizan las nuevas funcionalidades de Facebook para conectarse con nuestra aplicación, lo que es un muy buen resultado.

A continuación observamos el número de llamadas diarias a la API de Facebook [Figura 32], que se corresponde con el uso de las nuevas funcionalidades, como extraer a los amigos del usuario de Facebook.



Figura 31: Llamadas diarias a la API de Facebook

Como se puede observar en el gráfico, la media de llamadas realizadas diariamente a la APP de Facebook ha aumentado ligeramente, al igual que en el caso anterior. Cabe destacar que este aumento de llamadas no significa que los usuarios usen un mayor número de funcionalidades dentro de la aplicación, sino que el 20% de usuarios que han utilizado la identificación mediante Facebook, también acceden al resto de funcionalidades relacionadas con dicha web social.

Por último podemos observar la tasa de error obtenida al realizar llamadas a la API de Facebook **[Figura 33]**:

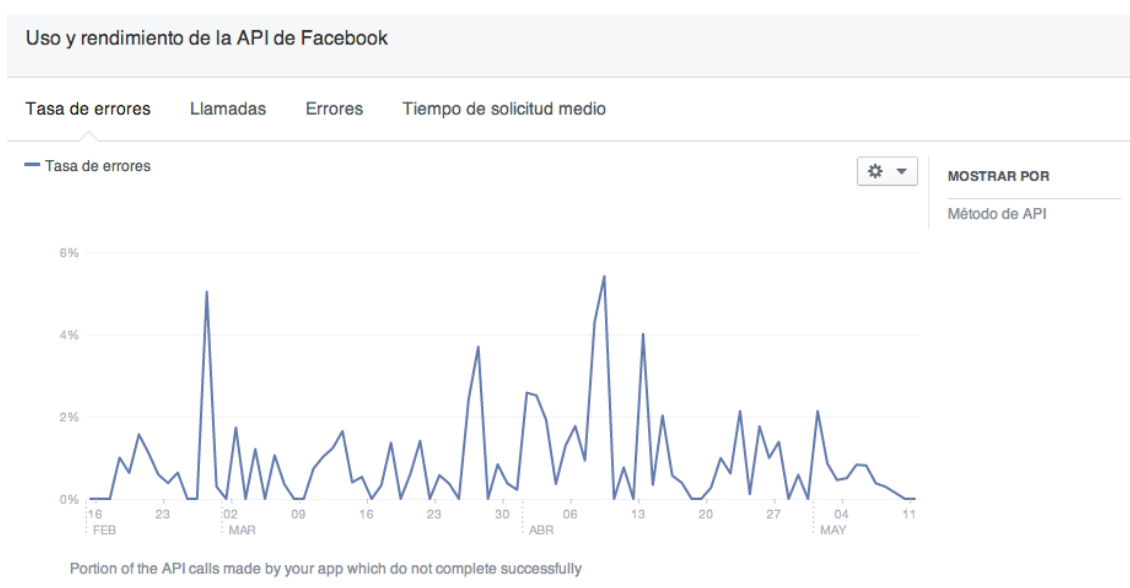


Figura 32: Tasa de error de las llamadas a la API

Dicha tasa de error se encuentra en todo momento por debajo de un 6%, sin embargo la media de la misma es de un 1%, únicamente superado en algunos casos atípicos, relacionados con la introducción de nueva funcionalidad al sistema, lo cual es un resultado más que aceptable.

En cuanto a las funcionalidades relacionadas con la intervención de usuarios, aún no se han podido extraer pruebas concluyentes, ya que se realizó la integración el 20 de Mayo, lo que supone un tiempo demasiado pequeño como para percibir una respuesta de los usuarios. Por otro lado es complicado asociar un cambio (tanto positivo como negativo) por parte de los usuarios a la implementación de estas funcionalidades, ya que influyen muchos otros factores en el comportamiento de los mismos.

5 CONCLUSIONES Y LÍNEAS FUTURAS

Durante el desarrollo de este proyecto se han analizado y llevado a cabo las tareas necesarias para aumentar la participación de los usuarios de un servicio web. Tras la implementación de las mismas se ha apreciado una mejora considerable en el número de usuarios que utilizan las nuevas funcionalidades implementadas.

Este proyecto ha aportado numerosos beneficios tanto a la empresa que oferta dicho servicio web como al alumno que ha llevado a cabo el mismo. Los cambios más significativos en el servicio, ya mencionados en el apartado anterior, han sido el aumento el número de usuarios que utilizan nuevas funcionalidades como la integración con Facebook y la mejora del tipo de usuario que menos atención ha tenido.

Gracias a estas mejoras se ha abierto un gran elenco de posibilidades en cuanto al trabajo futuro que aumentaría el valor de este servicio web. El **trabajo propuesto** para llevar a cabo en un corto o medio plazo es el siguiente:

- **Compartición de fotos con Facebook:** Requisito funcional propuesto, ya mencionado en el apartado 3.2 de este documento.
- **Creación de un sistema de etiquetas** para identificar a los integrantes de las fotos e importación de las mismas desde Facebook.
- **Ampliación de las Facebook Stories** para que todos los usuarios puedan compartir de forma automática los eventos acaecidos dentro de la aplicación y no únicamente la pareja.
- **Integración de otras redes** como Google+ o Instagram.
- Creación de un **ránking de usuarios** que mejore la participación en el evento.

Por otro lado, los **beneficios obtenidos de cara al alumno** son los siguientes:

- Ha **consolidado el conocimiento adquirido** de los siguientes lenguajes de programación utilizados en este proyecto: **PHP, AJAX, Javascript, SQL**. También ha utilizado el conocimiento adquirido en la carrera sobre **servidores y bases de datos**, además del **uso de patrones y algoritmos** aprendidos en la misma.
- Ha podido **formar parte de un equipo de desarrollo real** y aprendido a utilizar las **herramientas de organización y compartición de código**, como **JIRA** o **Subversion**. También ha aprendido cómo es el proceso de desarrollo llevado a cabo por una empresa y participado activamente en el mismo. Además, ha podido participar además en dinámicas para mejorar el trabajo en equipo de un grupo y enseñar lo aprendido durante este proyecto a otros integrantes del equipo.
- Ha realizado **un estudio completo de la API que ofrece Facebook**, lo cual será de utilidad para futuros trabajos llevados a cabo por el estudiante.

Por último, el trabajo desarrollado servirá de ejemplo a la hora de implementar futuras funcionalidades relacionadas con la API de Facebook, ya que el propósito ha sido implementar un buen producto que sirva de referencia para un trabajo futuro.

6 BIBLIOGRAFÍA

- | | | |
|------|---|------------|
| [1] | www.icasai.com.br | 20/05/2014 |
| [2] | www.series.ly | 20/05/2014 |
| [3] | www.pinterest.com | 20/05/2014 |
| [4] | www.goodreads.com | 20/05/2014 |
| [5] | https://developers.facebook.com/docs/ | 20/05/2014 |
| [6] | https://dev.twitter.com/docs/twitter-for-websites | 20/05/2014 |
| [7] | http://avatars.io | 20/05/2014 |
| [8] | http://www.zankyou.com | 20/05/2014 |
| [9] | https://developers.facebook.com/docs/opengraph/submission-process | 20/05/2014 |
| [10] | https://developers.facebook.com/docs/facebook-login/checklist | 20/05/2014 |
| [11] | http://framework.zend.com/manual/1.10/en/coding-standard.html | 20/05/2014 |
| [12] | https://developers.facebook.com/docs/facebook-login/access-tokens | 20/05/2014 |
| [13] | https://developers.facebook.com/docs/plugins/share-button | 20/05/2014 |
| [14] | https://developers.facebook.com/docs/opengraph/creating-custom-stories | 20/05/2014 |
| [15] | https://developers.facebook.com/docs/plugins/like-button | 20/05/2014 |

7 BIBLIOGRAFÍA COMPLEMENTARIA

Kerpen, Dave. *Likeable Social Media*. [Libro en línea] Summer 2011. [Washington, DC]: Library of Congress Cataloging-in-Publication Data [Consulta 20-5-2014] ISBN e-book 9780071762342

